

Effective Anchoring of Multiview Narrative Generation

Khyathi Raghavi Chandu

CMU-LTI-21-022

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213
www.lti.cs.cmu.edu

Thesis committee:

Alan W Black (Chair) (Carnegie Mellon University)
Eric Nyberg (Carnegie Mellon University)
Abhinav Gupta (Carnegie Mellon University)
Devi Parikh, Georgia Tech & Facebook AI Research

*Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
In Language and Information Technologies*

©2021, Khyathi Raghavi Chandu

Dedicated to my parents

Abstract

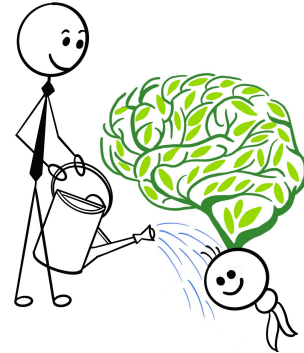
Humans inherently learn from and interact with multiple views of information, be it various modalities or languages. So, the expectations from contemporary and 21st-century technology are a testimony for the increasing need to model these multiview contexts better. Natural language generation plays a pivotal role in communicating these contexts in human-understandable languages. This thesis brings together both of these transformative technologies to make strides towards a longstanding dream of human-like multiview narrative generation. The critical challenge is identifying the natural-sounding properties of long-form texts and modeling them in tandem with visual contexts. This thesis presents *anchors for grounding three such properties* including content (relevance), structure (coherence), and surface form realization (expression), and anchors them with relevant visual contexts. These anchors also provide us with human interpretable handles for controlling these properties. Illustrating the effectiveness of the anchors for each of the three properties, I present:

Starting with *content*: In situated multimodal contexts, relevance is the concept of the elements in one modality being connected to the other modality that makes this context informative and complementary. I present visual infilling with curriculum learning as a global objective for content and hierarchically attending over entity skeletons as a local objective for content, to generate visual stories and procedures. To improve the controllability and transferability in English and five other languages, I also introduce a dual-stage model with weakly supervised skeletons and a text-as-side attention mechanism to denoise the content in an image caption. Moving onto *structure*: The alignment of description in language to the corresponding visual inputs is crucial to generate a logical and coherent narrative. I present a scaffolding technique as a local objective for structure by extracting a layout from vast amounts of unsupervised text to incorporate structure into cooking recipes generated from images. Finally *surface form*: The crux of naturalness to automatic generation comes by incorporating individualized and personalized ways of expressing the same content. I present a locally guided weakly supervised model for generating persona-based visual stories and a dual-staged adversarial technique to generate mixed view language from non-parallel data.

All the above work mainly focuses on static multimodal narratives, and I present a case to highlight the significance of transitioning to dynamic grounding. I conclude by presenting the shortcomings of the current approaches in the NLP domain to the grounding problem and offer recommendations along with executable actions for course correction to bridge this gap and enable grounding for machines to resemble human communication.

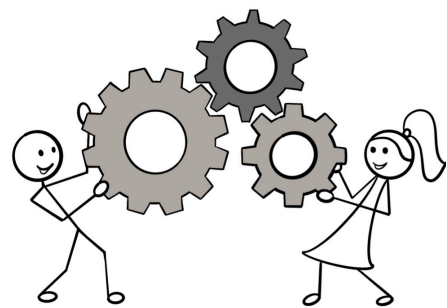
Acknowledgements

First and foremost, I would like to thank my advisor Alan W Black for your overwhelming support and guidance, without whom this thesis would not have been possible. Your scholarly insights enabled me to explore a rich spectrum of breadth and depth in my research. I am simply perpetually grateful to you for being a role model and nurturing my ability to grow as not only a researcher but also a well-rounded person. I could not have asked for a more positive influence in the beginning stages of my decision-making adulthood. I am in reverence of your mastery in treading



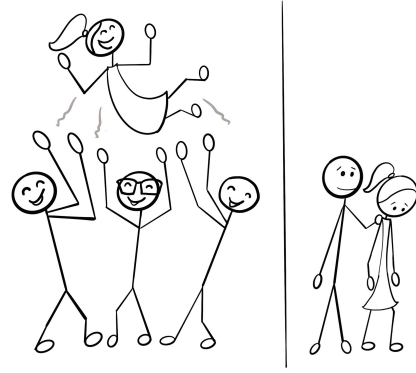
with poise on balancing between passion for work and compassion towards students. You have not only taught me but also taught me how to teach with empathy. Your intellectual inquisitiveness to learn new things, solicitude and always being there is contagious and I am grateful to absorb some of it. It never ceases to amaze me how much I learn from casual conversations with you. I would like to express my sincere and profound gratitude to my co-advisor, Eric Nyberg, and I am immensely grateful that you introduced me to a gamut of research skills. Thank you so much for giving me the freedom to explore new and challenging paths, which enriched my PhD experience. Your foresight in invention, passion for delivering the forefront of usable systems, and incredible managerial skills will be a constant inspiration throughout. I express my deepest gratitude to my committee members – Abhinav Gupta and Devi Parikh. Abhinav, thank you for teaching me the complementary techniques from vision to approach multimodal and the insightful questions and feedback that helped shape this thesis. Devi, your dedication to work and innovative research is an inspiration for me. Thank you so much for your prudent observations and advice that helped clarify and shape this thesis better.

I would like to earnestly express my gratitude to Yonatan Bisk for encouraging me to be bold with my ideas and teaching me to pre-think about research problems. I would like to thank all my collaborators and intern mentors for having the immense passion and patience for working with me on various projects. A special thanks to Piyush Sharma, Soravit Changpinyo, Ashish Thapliyal, Radu Soricut for the ardent brainstorming sessions and technical guidance. A special thanks to Piyush for patiently mentoring me and enlightening me about the various aspects of industry and professional career. Finding the right gear to

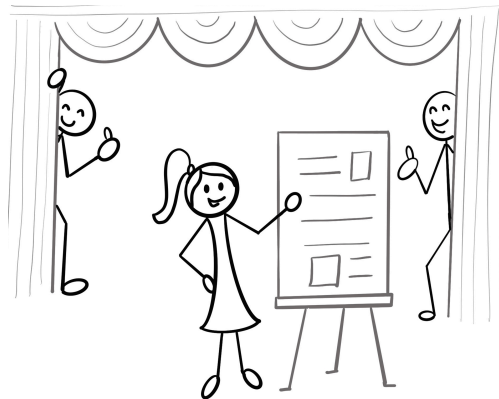


steer long-term outcomes for short-term research goals is achieved through teamwork. I am incredibly grateful to share this team space with my collaborators Sunayana Sitaram, Manoj Chinnakotla, Shrimai Prabhumoye, Ruo-Ping Dong, Thomas Manzini, Sai Krishna Rallabandi, Sumeet Singh for making the projects more engaging and enjoyable.

I would like to thank my labmates, Sai Krishna and Sanket Mehta, and my roommate Aakanksha Naik. From the countless whiteboard technical derivations, late-night philosophical discussions, you have been a stained-glass window in cultivating and connecting dots of not only research but all dimensions of life. I am grateful to be surrounded by friends who have been there with me in uncertainties and joined my joy. Thank you Abhijit Gopakumar, Danish Pruthi, Mansi Gupta, Siddharth Dalmia, Abhilasha Ravichander, Jonathan Francis, Aditya Chandrasekar, Shruti Rijhwani, Harsh Jhamtani, Bhuwan Dhingra, Kundan Krishna, Tanmay Parekh, Thomas Manzini, Shruti Palaskar, Sahitya Potluri and many more, for inspiring me in small everyday things you do and the amazing things you achieve. My special thanks go to Abhishek for being the cheerful constant through the crests and troughs. Thank you for always believing in me and being my treasured latent energy source.



I would like to thank my parents, who are my silent heroes behind the curtains, always hearteningly supportive of all my inclinations and aspirations, including my journey throughout this PhD. Your unconditional love truly encompassed me, and I feel blessed to be your daughter. Thank you for leading by example in pursuing my dreams, and more importantly, instilling the value system of work ethics. I am extremely fortunate to have my sisters, Sharvani and Gayathri, and for all your care and encouragement throughout. Thank you for sharing a lifetime of love and support.



Finally, I would like to thank all the people whose work and thoughts positively influence me. Your interviews, papers, and blogs have been tremendously inspirational to me. And I would like to mention a huge thanks to all the readers interested in my work.

Contents

Abstract	i
Acknowledgements	ii
List of Figures	vii
List of Tables	x
1 Introduction	1
1.1 What makes a narrative compelling ?	2
1.2 Thesis Statement	4
1.3 Thesis Overview	4
I Background	8
2 Anchoring and Neural Text Generation	9
2.1 Anchoring Framework for Generation	9
2.1.1 Motivations for Anchoring	12
2.1.2 Typology of Narratives	13
2.2 Task Agnostic Text Generation Techniques	16
2.2.1 Training Paradigms	17
2.2.2 Core Modeling	19
2.2.3 Decoding Strategies	20
2.2.4 Key Challenges	22
2.2.5 Evaluation	25
2.3 Modeling Approaches for Generation and Anchoring	27
2.3.1 Anchored Text Generation	32
2.4 Multiview Interactions with Anchors	36
2.5 Conclusions	37

II	Anchoring Narrative Properties	39
3	Content in Narratives: N-Local and N-Global Anchoring	40
3.1	Related Work	41
3.2	Content selection for Denoising Image Captioning	45
3.2.1	Datasets Description	46
3.2.2	Models Description	47
3.2.3	Experiments and Results	50
3.3	N-Local Anchoring from Entities	59
3.3.1	Dataset Description	60
3.3.2	Models Description	61
3.3.3	Experiments and Results	68
3.4	N-Local Anchoring from Implicit Contexts	71
3.4.1	Dataset Description	73
3.4.2	Models Description	75
3.4.3	Experiments and Results	78
3.5	N-Global Anchoring from Questions	80
3.5.1	Dataset Description	81
3.5.2	Model Description	82
3.5.3	Experiments and Results	88
3.6	Conclusions and Prospective Future Directions	90
4	Structure in Narratives: N-Local and N-Global Anchoring	94
4.1	Related Work	96
4.2	N-Local Anchoring from phases in recipes	97
4.2.1	Data Collection and Description	98
4.2.2	Models Description	99
4.2.3	N-Local Anchor Representation: Phases and States	102
4.2.4	Experiments and Results	102
4.3	N-Global Anchoring from Reordering	104
4.3.1	Models Description	105
4.3.2	Analysis of Ordering	108
4.3.3	Experiments and Results	111
4.4	Conclusions and Prospective Future Directions	115
5	Surface Form Realization: N-Local and N-Global Anchoring	117
5.1	Related Work	118
5.2	N-Local Anchoring in Language Information	122

5.2.1	N-Local Anchoring for Language Modeling	122
5.2.2	N-Local Anchoring for Speech Synthesis:	131
5.3	N-Local Anchoring in Personality	137
5.3.1	Datasets Description	139
5.3.2	Model Description	140
5.3.3	Experiments and Results	144
5.4	N-Global Anchoring in Language Information	148
5.4.1	Datasets Description	149
5.4.2	Model Description	149
5.4.3	Experiments and Results	153
5.5	Conclusions and Prospective Future Directions	155
 III Looking Forward		158
 6 Grounding ‘Grounding’ in NLP		159
6.1	Dimensions of grounding	160
6.1.1	Dimension 1: Coordination in grounding	160
6.1.2	Dimension 2: Purviews of grounding	162
6.1.3	Dimension 3: Constraints of grounding	163
6.2	Grounding ‘Grounding’	164
6.2.1	Data and Annotations	165
6.2.2	Domains of grounding	165
6.2.3	Approaches to grounding	165
6.3	Analysis of trends	171
6.4	Path Ahead: Towards New Tasks and Repurposing Existing Datasets	173
 7 Known Unknowns		175
7.1	Summary of Contributions	175
7.2	Future Directions	177
7.3	Broader Impact	185
7.4	Ethical Considerations	192
 Bibliography		198

List of Figures

1.1	Examples of manifestation of <i>content</i> , <i>structure</i> and <i>surface-forms</i> in a story-like text and instructive text.	2
1.2	Data Flow Diagram for Anchoring in Generation Model	4
2.1	Overview of N-Local Anchoring.	11
2.2	Overview of N-Global Anchoring.	11
2.3	Overview of Anchoring in Narrative Properties for Generation	12
2.4	Outlining the components of neural text generation discussed through the section.	17
2.5	Interactions among multiple views (visual modality with textual anchors) . . .	37
3.1	Overview of our approach: (1) skeleton prediction & (2) skeleton based IC; compared to conventional IC. Output captions shown in English (En), Hindi (Hi) and Italian (It).	45
3.2	Model architecture of our skeleton based captioning along with <i>text as side attention</i> mechanism between visual (v) and textual (w) modalities. The skeleton is present optionally in the encoder, decoder or both based on our three approaches.	47
3.3	Comparison of the content-anchoring based on multimodal interactions in denoising	50
3.4	Captions generated by baseline and our dual staged approach in 6 languages and their corresponding translations.	52
3.5	Human evaluation interface: We ask raters to: 1) compare the two captions (relative), 2) give ratings for each caption (absolute).	53
3.6	Controllability: Effect of guiding the information through the skeleton. As observed, the caption incorporates information from the skeleton that is consistent with the image. For example, we see that peace is incorporated in the second column of the top row while harbor and heaven are not. The relevant skeleton words in other columns guide the captions accordingly.	54
3.7	Quantitative relationship between the number of skeleton words and caption length.	58
3.8	Controllability: Effect of varying the number of words in the skeleton on the generated caption length.	58
3.9	Architecture of Glocal Hierarchical Attention on Entity Anchors with coreference chains to perform Visual Storytelling	65
3.10	Qualitative Analysis	66
3.11	Visualization of the hierarchically attended representation of the skeleton for story in Figure 3.10	67
3.12	Comparison of the content-anchoring based on multimodal interactions for modeling entities and coreferences	68

3.13	Percentage of Entities in the form of Nouns and Pronouns in the generated stories	70
3.14	Overview of infilling in visual procedures. Image in the second step is masked while the model generates the corresponding textual description from surrounding context.	71
3.15	Comparison of the content-anchoring based on multimodal interactions in infilling	77
3.16	Comparison of V-Infill and XE dealing with infilling context during inference (for making <i>chicken roast</i>). GT corresponds to the ground truth step. The index in each row corresponds to the index of the missing image.	78
3.17	Human Evaluation Interface for an example of generated recipes with both techniques.	80
3.18	System pipeline for Ideal Answer Generation (with configuration choices) . .	83
3.19	Summaries generated with different techniques	89
4.1	Storyboard for the recipe of vegetable lasagna	98
4.2	Architecture for incorporating high level structure in neural recipe generation by n-local anchoring through state machines	99
4.3	Comparison of the structure-anchoring based on multimodal interactions . . .	103
4.4	Comparison of generated <i>storyboards</i> for <i>Easy Oven Baked Crispy Chicken Wings</i> . .	103
4.5	(a) Similarity Ordering (b) Majority Ordering (c) Block Ordering	107
4.6	Quality of Fused Sentences	113
5.1	Anchoring language id units in Language Modeling of Code-Switching	125
5.2	Comparison of the surface-form-anchoring based on lid for language modeling	128
5.3	Sample Code-Mixed generation by AWD-LSTM along with gloss and translation (text in red are Hindi words that weren't identified correctly in the LID step. . .	129
5.4	English Word Embeddings	130
5.5	Hindi Word Embeddings	130
5.6	Architecture of the system with example of Hindi navigation instruction (Note that the language of the word 'Chowk' is misidentified and transliteration of 'karawal' is incorrect)	133
5.7	Comparison of the surface-form-anchoring based on lid for speech synthesis .	135
5.8	Comparison of the surface-form-anchoring based on multimodal interactions for persona based generation	145
5.9	Comparison of generated <i>stories</i> from all the described models.	146
5.10	Transformer based GAN architecture for generating CS text. <i>Note: The same architecture is used for two stages. Matrix language sentence is the embedding of the text and language encoding is the embedding of the language.</i>	151
5.11	Comparison of the surface-form-anchoring based on discriminator for language generation	152
5.12	Trends in metrics for evaluating the generation of CS text for four language pairs in dual stage training. <i>The dotted line of each color benchmarks the corresponding metric for real CS data.</i>	153
6.1	Dimensions of grounding – required to bridge the gap between current state of research and what is missing from grounding in real sense.	160
6.2	Coordination in grounding	162
6.3	Approaches to grounding	166

6.4	Analysis on the trends in grounding	172
6.5	Analysis of Domains and Techniques	173

List of Tables

1.1	Organization of thesis	5
2.1	Paradigms of Tasks in Text Generation (not detailed in this chapter). Note: To be compact, we include ‘Knowledge-to-text’ paradigm within ‘Data-to-text’.	18
3.1	The inputs and outputs of the different models. In iterative refinement, S' is replaced by C'	49
3.2	Performance of skeleton prediction stage. Note that for classification and generation, the skeleton type used is ‘nouns & verbs’.	51
3.3	Automatic metrics to compare various skeleton forms. <i>Img2Cap</i> is the baseline (<i>large</i> version refers to 12 encoder and decoder layers). Note that these results use generation-based skeleton prediction.	51
3.4	CIDEr scores for skeleton (form: Nouns & Verbs, prediction approach: generation) conditioned caption generation for multiple languages.	52
3.5	Ablations on val data for unpaired captioning.	52
3.6	Human evaluation scores of different approaches and skeletons on English (vs the <i>Img2Cap</i> baseline).	54
3.7	Human evaluation results for skeleton (form: nouns & verbs, prediction approach: generation) conditioned caption generation for multiple languages.	54
3.8	Analysis of multimodal discourse coherence relations for baseline and our model on T2 dataset. The last column shows the relative human evaluation gains over baseline caption of each type. Other relations with small counts are ignored in the above analysis.	55
3.9	Human evaluation results on <i>SkeEnc</i> model for skeleton (form: nouns & verbs, prediction approach: generation) conditioned caption generation for multiple languages.	55
3.10	Human evaluation results of comparison between the generation and classification based approaches	56
3.11	Absolute ratings in percentages in Human Evaluations.	56
3.12	Details of the ViST Dataset	61
3.13	Examples of three forms of Entity-Coreference Schema Representation for representing n-local anchors	62
3.14	Automatic Evaluation of Story Generation Models	68
3.15	Details of the ViST and <i>Visual Procedure Telling</i> Dataset broken down into 10 categories	72
3.16	Regrouping the categories in ViPT dataset	74
3.17	Performance of different models on stories (from ViST) and recipes (from ViPT) datasets	75
3.18	Performance of infilling during inference for recipes in <i>Visual Procedure Telling</i>	75
3.19	Performance of infilling during inference for <i>Visual Story Telling</i>	75

3.20	Compatibility of anchor form representation with similarity metrics	86
3.21	ROUGE scores with different algorithms, ontologies and similarity metrics . .	88
4.1	Details of dataset for <i>storyboarding</i> recipes	99
4.2	BLEU Scores for different number of phases (\mathbb{P}) and states(\mathbb{S})	102
4.3	Evaluation of storyboarding recipes	103
4.4	Manual evaluation of sentence ordering	108
4.5	Performance of different module combinations on Test Batch 4, BioASQ 4th edition; R=Rouge, Pr=Precision, Re=Recall	111
4.6	System performance comparing Fusion + Ordering and Ordering + Fusion . .	114
5.1	Hinglish Data Statistics for Code-Switched Language Modeling	124
5.2	Perplexity scores of different models from Anchoring Language Ids	129
5.3	Navigation Instructions: Data Statistics	132
5.4	Subjective listening tests for preference in synthesis of mixed-language navigational instructions	136
5.5	Subjective listening tests with drivers for synthesis of mixed language navigational instructions	137
5.6	Statistics of data belonging to each of the persona clusters	145
5.7	Performance of classifiers for each of the persona clusters	145
5.8	Performance (in terms of accuracy) of generated stories to capture persona . .	146
5.9	ROUGE_L scores for the generated stories by each of our models	147
5.10	Monolingual and Code-Switched Datasets used for training Stage 1 and Stage 2	150
6.1	Pointers to curated datasets introduced to address <i>grounding</i>	166

1

Introduction

Narration and storytelling emerge as a natural corollary of innovation.

Free-form narrative intelligence has been a long-standing dream for artificial intelligence. This uniquely human ability has evolved over generations from cave paintings to novels to digital media, varying from social media posts to a series of movies. The ancient tradition of humans using shadows against fire is paralleled to using artificial intelligence techniques evolved from technology to depict stories. The ability to perceive and express a set of related events in a coherent narrative is an innate human attribute. With the advent of virtual beings co-existing and sharing the ecosystem with us, it is time to teach them this skill. This reverberates the necessity of *computational narrative intelligence* from traditional standpoints (Schank, 1990; Opt, 1988; Brewer, 1982; Lehnert and Vine, 1987; Lehnert, 1982) to more contemporary approaches and applications (Ouyang and McKeown, 2015; Murray, 2015; Riedl and Harrison, 2016; Harrison and Riedl, 2016; Chaturvedi et al., 2018; Underwood et al.). Narratives represent a shared cohesive understanding. Hence they are a powerful means to shape our minds and decisions. These can range from a lawyer's courtroom argument (Delgado, 1989) to online advertising that determines the attitude towards a product (Chang, 2009; Ching et al., 2013). Transcending to the conceptual dream of interacting with our virtual assistants in such narratives can be very beneficial. Hence, strategizing their functionalities and roles is imperative to maximize these benefits. Maintaining this overarching goal in mind, we will first see the contributing factors that make a narrative effective.

1.1 What makes a narrative compelling ?

Narrative/Multi-sentence Generation: A narrative is a cohesion of more than one sentence in conjunction. Several efforts have been made for single sentence generation like machine translation, response generation, paraphrasing, etc. While this is essential and practically exceedingly viable, the same techniques are not effective for *multi-sentence or long-form generation*. In this thesis, I mainly focus on long-form generation in the domains of stories and procedures (or instructions) ¹.

In this thesis, I work towards three of the critical properties or dimensions to emulate the competence of humans in generation by artificial agents. They are defined as follows:

- **Content:** It is the quality of the text being closely connected, appropriate and informative. Thus, it contributes to improving the *relevance* of the text.
- **Structure:** It is the quality of the text being logical and consistent following a structure. Thus, it contributes to improving the *coherence* of the text.
- **Surface-form realization:** It is the choice of the surface tokens derived from underlying representation to generate the actual text. Thus, it contributes to the *expression* of the text.

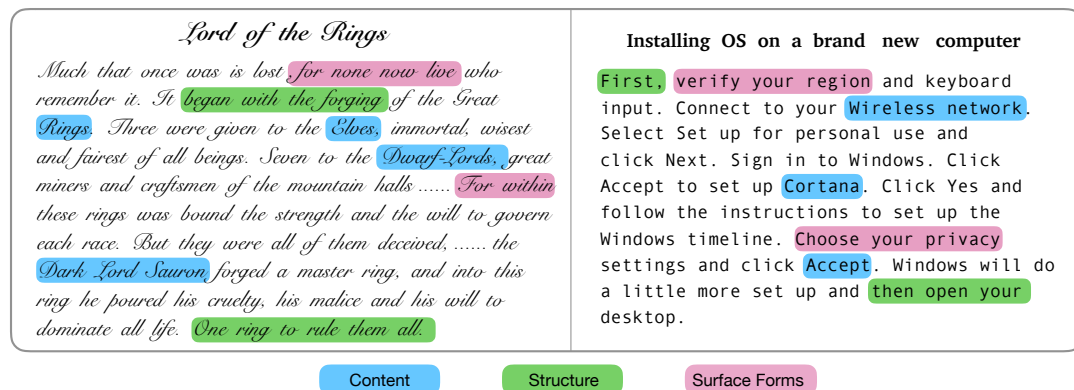


FIGURE 1.1: Examples of manifestation of *content*, *structure* and *surface-forms* in a story-like text and instructive text.

Examples of these properties are demonstrated in Figure 1.1 which include snippets from ‘Lord of the Rings’ and ‘Installing OS on a brand new computer’. While the former is picked up from a story, the latter is a ‘how-to’ activity. As observed from the figure, the three properties discussed above are manifested in sufficiently distinct ways to suit the intent and the context of the narratives. The choice of words contributing to the **content** are very domain-specific like ‘Elves’, ‘Dwarf-Lords’ in the story and ‘Wireless’ and ‘Cortana’ on the right. The **structural** layout on the left uses ‘it began with the forging’ and concludes at the end. Similarly, on the right, this is indicated in the form of a step-by-step procedure where it begins with cues like ‘first’ and finally concludes with the words like ‘then open your’. The tone of the presentation

¹Other popularly studied multi-sentence generation include summarization, story generation, etc.,

or **surface** form realization has an archaic style with the use of phrases like *'for none now live'* in the story and instructive or imperative sentences with phrases like *'choose your privacy'* for how-to texts. Fundamentally, these are the three properties of narratives that we focus on improving in this thesis.

We carry around our personal assistants such as *Siri, Alexa, Google Assistant* capable of carrying out full-fledged yet independent tasks such as booking flights, reserving appointments etc., In addition to these current capabilities, these virtual assistants also need to pick up on the right cues to bring forth the right kind of narratives and present it based on relevance to an audience from their clairvoyant faculties. The skill of incorporating the aforementioned narrative properties comes naturally to humans, and we seek to impart it to them. We have explored the various constituency properties that make a narrative effective. In the same way that we are enveloped in narratives in our surroundings, the context encompasses multiple modalities and multiple languages. This brings forth an additional dimension of embedding these properties in situated contexts. In this thesis, I scope these narrative properties in multi-modal and multilingual scenarios. In specific, we will be looking into the modalities of *vision* and *language*. Accruing to these views, these personal assistants can be revolutionized with capabilities serving virtual AI characters embedded in Augmented Reality, Virtual Reality and Internet of Things that can cater to a wider range of responsibilities such as digital education, entertainment, personal counseling etc.,

Improving multi-modal and multi-lingual capabilities: Humans interact and engage with different modalities in regular communication. With the growing ubiquity of various media and modes used to share information, including videos and audio, incorporating multimodality and multilinguality is paramount. In this thesis, I work on visual, auditory, and textual modalities as input, focusing on generating text. A significant portion of the work delves into capturing the aforementioned narrative properties in text from a sequence of visual information. In addition, the work also describes extending text generation from one language to a mix of multiple languages.

Textual Anchors and Visual Inputs: This thesis specifically focuses on anchors from textual modality modeled with visual input to generate long form texts. This data flow of this process is demonstrated in Figure 1.2.

First, we use the paired or unpaired text to extract anchors. Since these anchors are retrieved from text, the anchors are textual as well. Based on the target narrative property that we are trying to improve, we extract the corresponding anchors for content, structure and surface forms. These textual anchors along with the input images are used by the anchored generation model to generate narrative texts. To efficiently bring together multiple modalities or languages and narrative generation, I present categories of modeling these anchors in the next chapter.

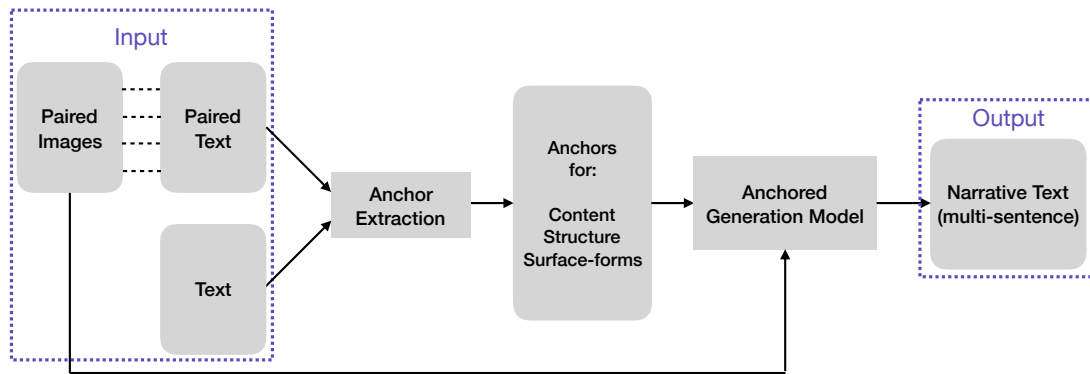


FIGURE 1.2: Data Flow Diagram for Anchoring in Generation Model

1.2 Thesis Statement

The central tenet of this thesis is to improve long-form text generation by anchoring them to aspects of human communication, especially in multimodal contexts. Anthropomorphic narrative generation in natural language via stories, procedures, etc., has been a long-standing dream of artificial intelligence. Working towards this goal brings forth the need to adhere to the innate human characteristics of narratives. They include content (relevance), structure (coherence), and surface form realization (expression). Anchoring these narrative properties is maneuvered not only by task-specific requirements but also by the availability of annotated data. A steep acceleration in brewing new content every day both surmounts and impedes the need for extensive annotations. The main contribution of this thesis is a two-dimensional taxonomy of anchoring these three properties by locally and globally conditioned training objectives. This framework taps into techniques for anchoring to improve narrative generation. For anchoring content, I introduce anchor biased attention model to denoise, improve controllability and cross-lingual transferability. I also introduce a hierarchical entity model to learn where and how to introduce entity words and an infilling-based model to address the real world problem of missing contexts and thereby improve robustness. For anchoring structure, I present a scaffolding structure representation from images for text generation and reordering sentences. For anchoring surface-form realization, I present solutions with adversarial training and multi-tasking. Finally, I circumspect the limitations in the assumptions of ongoing work on grounding and present missing dimensions along with potential paths to bridge this gap.

1.3 Thesis Overview

Anchoring framework and background techniques [Chapter 2]: Long text narratives often comprise implicit and explicit building blocks to make them more compelling and readable. First, I formally define the n-local or n-global anchors in §2.1 followed by the modeling and task based motivations for this framework in §2.1.1. Then, I present a background of task-agnostic techniques that are used for grounding these properties §2.2. The narrative properties that make it efficient are grounded in different levels of granularity of these anchors. Following this, since most of this thesis derives anchors from textual modality, I present categories

<i>Narrative Properties</i>	<i>N-Local Anchors</i>	<i>N-Global Anchors</i>
Content	Entities, character references, missing contexts	Query words
Structure	Unsupervised structural layouts	Sentence order and fusion
Surface forms	Fixed language ids	Flexible language ids

TABLE 1.1: Organization of thesis

of methods to club visual input with text anchors. After introducing these building blocks for anchoring formally, I present an overview of modeling approaches and tasks for anchored generation in §2.3. Finally, I present categories of approaches for multimodal anchoring in §2.4.

The aforementioned narrative properties and anchor points can be organized based on their granularity in driving the narrative into the following segregation. This is depicted in Table 1.1. This layout brings forth three main chapters in this thesis post introduction. Each of the chapters addresses a narrative property from the perspective of both n-local and n-global anchor points.

Anchored in Content [Chapter 3]: Anchoring the narrative in appropriate content helps improve the relevance of the text (and thereby implicitly a degree of relevance and coherence). In this chapter, I present methods to anchor entities, references, query words and missing contexts in the narratives.

- *N-Local Anchors:* Entities are widely used to represent the content in a story as characters involved in various plot points. I first present a method to utilize anchors to *denoise*, *control* and *transfer* content appropriately. I then present methods to utilize reference forms to improve the content in visual stories. These content-based anchor points from entities and their references are extracted to form skeletons. Based on their properties, different forms of these skeletons are used to generate a story from a sequence of images. Every sentence is undeviatingly guided based on the entities present in the corresponding image. Alternative to explicit provision of content, I propose a model that infills the relevant content from surrounding contexts via curriculum learning. This *local* incorporation of anchor units, i.e., entities or references, determines the content in every sentence at fine-grained level.
- *N-Global Anchors:* The content in a summarized answer for a query is driven by a theme or the topic being discussed in the query. This content is determined using surface form lexical units and the relevant ontologically related words in the question. The theme or the topic of the answer narrative is selected at the narrative level. This theme is observed in the overall summary to maintain a holistic consistency of the related content. Though this general theme is governed by the input, not every sentence is strictly or intricately maneuvered from the query. Balancing between selecting related content while avoiding repetition is of chief essence. Hence, conforming to the previously extracted content provides global guidance to select the subsequent sentences or content units.

Anchored in Structure [Chapter [Chapter 4]]: Humans inherently maintain and adhere to a structure while narrating to make logical exposition and sense of accounted events. While several factors such as the grammar and lexicon are essential, a compelling narrative also has a good introduction, detailing each portion, and finally ties these concepts up as a conclusion. Stringing these discrete units enforces a structure. This structuring can be inherently incorporated during learning or performed as a post-editing step after the content is accumulated. These are the two ways in which structure is used as anchor points.

- *N-Local Anchors:* I propose a method that learns structure representations in an unsupervised fashion. The first step is to derive weak annotations for the narratives at hand. The manifestation of structure is more apparent in procedural texts where the occurrence of a phase logically precedes/succeeds another phase. The structural anchors are derived based on the relationship between these amorphous phases. The explosion of state space emerging from raw forms is addressed using clustering. This reveals phase and state sequences that are incorporated at each step during training with a structure-based loss objective. The phase and the state sequence information is used to guide every step of the recipe granularly.
- *N-Global Anchors:* Controlling the layout of a narrative (in this case, a query-oriented summary) at a fine-grained level from a topic provided in a query is challenging. So instead, I work on a method to provide comprehensive guidance that is not entirely from the question but rather from a derivation or processed form. More concretely, the content accumulated from content-based anchoring now controls the restructuring of the sentences among themselves to arrive at a summary that smoothens transitions between different content units. This process is executed as a post-editing step.

Anchored in Surface Realization [Chapter 3]: The linguistic realization or lexicalization, often with discretion in surface form, presents a challenge for a style-based generation. It refers to communicative preferences defined as a set of linguistic variants based on social associations. It is often realized in the form of preferences based on stylistic, gender, political and cultural aspects. However, this thesis focuses on surface realizations derived from personas and interaction between multiple languages. Such interaction within a single utterance is also known as code-switching. Controlling the language information received during generation can anchor these surface forms.

- *N-Local Anchors:* A fixed sequence of lexical level language-ids can ground or anchor each word stringently to realize in one of the participating languages. However, this demands adhering to a sequence of languages during intermixing. Explicitly intricately providing language id tags is substantiated in language modeling of code-switched text in a multi-task learning framework to evaluate the significance of the n-local anchoring. Accompanied by this, we also present the synthesis of mixed language instructions and demonstrate that anchoring in n-local language ids designates improvement in the naturalness of the synthesis.
- *N-Global Anchors:* Surface form realization is often gauged in uni-dimensional and also resource-rich scenarios. To address this, I work on *n-global anchoring* of the surface

forms with limited annotations and non-parallel data. Instead of explicitly rendering the language id information mentioned earlier to each lexical unit individually, I propose a model that provides them at an abstract level for the entire sentence. I demonstrate experiments on learning these surface forms from monolingual data of the participating languages and argue that dual-stage training partially guides this mixed language information to generate code-switched text.

Bridging the gap in grounded generation [Chapter 6]: Human communication is more incidental than accidental, with circumstantially defined intents and grounded in context. In this chapter, I bring forth the dimensions of grounding critical to stride towards a more natural interaction, including dynamic grounding, purviews of extending grounding levels, and constraints imposed by the medium of communication. In this regard, I also sketch a path ahead with systematic solutions by defining a new paradigm for longitudinal benchmarking, and human-in-the-loop solutions.

Known Unknowns [Chapter 7] : In this chapter, I summarize and conclude the main facets of the thesis. Then, I briefly present prospective future directions, each accompanied by the known contexts (conjectured from this thesis), known background (brief related work), partially defined research question, and potential approaches that are unknown or unexplored experimentally.

This is the overall picture of my thesis, and we will delve deeper into each of the topics mentioned above in the following chapters. In the next chapter, a categorization of the prior work in some popular generation tasks is organized based on the anchoring framework along with task-agnostic techniques for generation.

Part I

Background

2

Anchoring and Neural Text Generation

Wise men speak because they have something to say; not to say something.

attributed to Plato

We cultivate language usage with a communicative intent or a goal (Bruner, 1974). Usually, this means that we actually have *something to say* that is *anchored* to an intent or a mind map, etc., The number of times that I rewrote and revised the details in this chapter indicates that there are various ways to gather my content, organize the structure and express my intent effectively. These several attempts are formulations of the anchors for my narrative or the *something* that I intend to communicate. While anchors ground language both in natural language understanding (NLU) and natural language generation (NLG), this thesis focuses on how they are used for generation. In this chapter, I first describe the anchoring framework and then provide a background of various task agnostic techniques for text generation.

2.1 Anchoring Framework for Generation

What is an anchor? *An anchor is a supporting framework, a basic structure, or a condensed essential primitive of something.*

This chapter is partly based on the following paper:

- “Positioning yourself in the maze of Neural Text Generation: A Task-Agnostic Survey” (Chandu and Black, 2020a)

An anchor is a reliable or principal form of support that the narrative is based on. The proposed framework characterizes guidance in two levels of granularity thereby introducing n-local and n-global anchoring. While this overarching goal of narrative generation aims at a multitude and diverse set of challenges, this thesis targets exploration of grounding narratives in anchor points. Situating these narratives by anchoring on real world contexts such as conversational cues, visual input etc, makes them more relevant, appurtenant and relatable.

Formalizing and realizing anchors within the scope of training procedures emerges from the degree of supervision needed to generate text, thereby varying from discrete to continuous anchors or fine-grained to coarse-grained anchors. Thus, manifesting them in the two ends of the spectrum results in two genera of anchoring: (i) N-Local anchoring and (ii) N-Global anchoring. Let us now take a closer look at each of these categories. Figure demonstrates the contrast between these two kinds of anchoring very broadly and we will discuss their formal definitions next.

As discussed in Chapter 1, a narrative is a sequence of multiple utterances about a connected sequence of events or topics. So, formally, let a narrative comprise a sequence of k units: $\mathbf{N}_i = \{\mathbf{s}_i^{(1)}, \mathbf{s}_i^{(2)}, \dots, \mathbf{s}_i^{(k)}\}$, where \mathbf{N}_i is the i^{th} narrative in the data with k units denoted by \mathbf{s} . Since these primitive units are primarily derived from text, they can be in the form of a word or a sentence or a sequence of sentences.

N-Local Anchoring: The training procedure optimizes for each unit of generation. As mentioned earlier, this unit can either be in the form of a word, a sentence, or a sequence of coherent sentences. The crucial aspect here is that each input unit in a sequence is paired with an anchor unit corresponding to one of the narrative properties (discussed in §1.1). This is depicted in the left hand side of Figure 2.1. On the right side, there is an overview of these anchors are modeled. The anchors are pairwise associated to the generated text and the dotted arrows from anchor units to each of encoder, model and decoder indicates the visibility of the modeling component into the anchors.

A formal representation of this category of anchors pertains to the definition of explicitly, elaborately and locally guiding each aspect of generation. Hence, the anchor is represented as $\mathbf{A}_i = \{\mathbf{a}_i^{(1)}, \mathbf{a}_i^{(2)}, \dots, \mathbf{a}_i^{(k)}\}$. These anchors can be derived from either of the participating modalities or languages from paired or unpaired text \mathbf{N}_i or visual input \mathbf{I}_i . Given \mathbf{I}_i and \mathbf{A}_i , the task is to generate \mathbf{N}_i .

Input: \mathbf{I}_i and $\mathbf{A}_i = \{\mathbf{a}_i^{(1)}, \mathbf{a}_i^{(2)}, \dots, \mathbf{a}_i^{(k)}\}$
Output: $\mathbf{N}_i = \{\mathbf{s}_i^{(1)}, \mathbf{s}_i^{(2)}, \dots, \mathbf{s}_i^{(k)}\}$

Note that the guidance is at a granular level as for $\mathbf{s}_i^{(j)}$, there is a corresponding anchor unit $\mathbf{a}_i^{(j)}$, where j is the index of the unit in generation. This is a supervised learning instance, since the anchoring is derived from annotations on the training data. Our focus in this thesis is not to improve the derivation of these anchor points and hence use off the shelf tools to procure these annotations. The anchors \mathbf{A}_i can take several forms facilitating the corresponding narrative property. This is one end of the spectrum where every time step in generation is anchored.

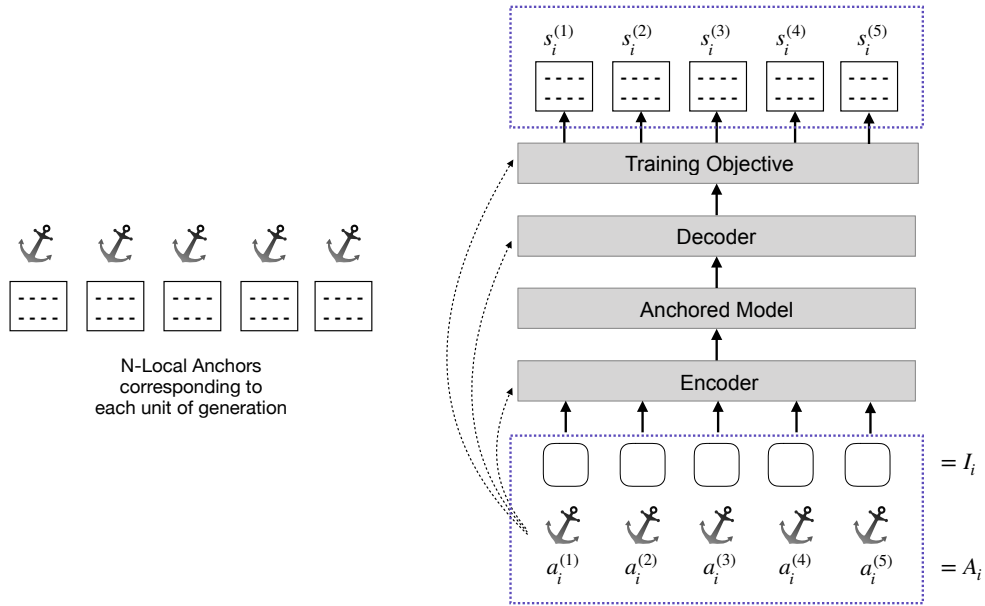


FIGURE 2.1: Overview of N-Local Anchoring.

N-Global Anchoring: The training procedure optimizes at the narrative level. This is demonstrated in Figure 2.2, implying that overall high-level guidance is provided for the entire narrative. The underlying anchor serves as a guiding theme and does not provide dense guidance for each narrative unit. Instead, a blanket proposition is served in the context of the narrative.

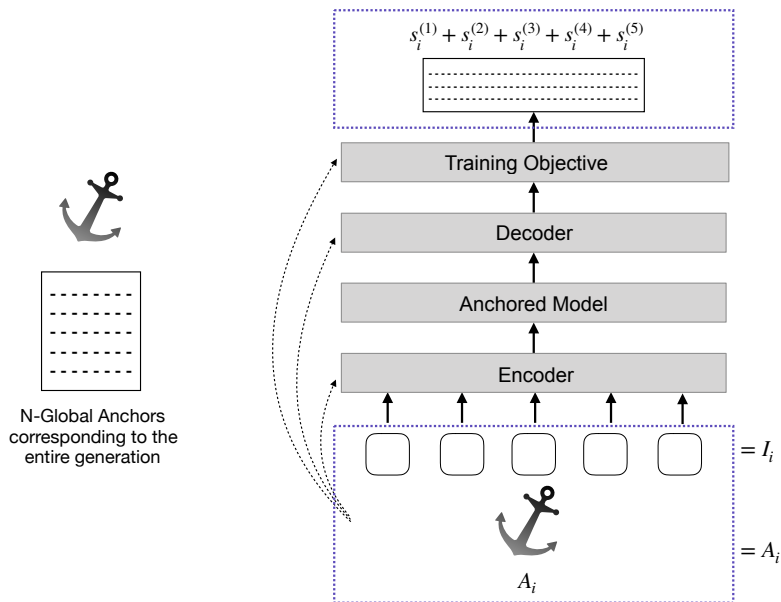


FIGURE 2.2: Overview of N-Global Anchoring.

Formally, this category of anchors compares with the definition where the anchors are provided in a coarser-grained detail. Hence the anchor is represented as A_i which does not contain atomic units granularly to guide the generation process. There is no one-to-one bijective

function from the anchor units to the generated text to be more concrete. Hence the entire narrative is generated at once with global guidance.

Input: I_i and an anchor A_i corresponding to the overall narrative

Output: $N_i = \{s_i^{(1)}, s_i^{(2)}, \dots, s_i^{(k)}\}$ guided by A_i .

I present an account of anchoring each of the narrative properties discussed in Chapter 1 (i.e., content, structure, and surface form realization) with N-Local and N-Global anchors. This overview is presented in Figure 2.3. Throughout this thesis, we are going to look into the epiphenomenon of making narratives effective by both sorts of anchoring into the narrative properties mentioned above.

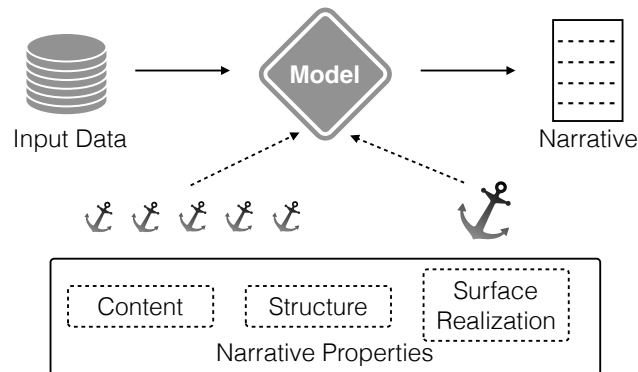


FIGURE 2.3: Overview of Anchoring in Narrative Properties for Generation

In addition to the formalizations of anchors for improving narrative generation, modeling-based and task-based requirements also assert the significance of anchor-based generation.

2.1.1 Motivations for Anchoring

The high-level motivation for anchoring is that humans inherently learn from and interact with multiple views of contexts, be it various modalities or languages. Therefore anchoring the contexts to these views is critical. In addition to this broad idea, this section presents two levels of motivations for this framework: modeling-based and task-based motivations.

Modeling-based Motivations: The 80's and 90's have witnessed significant efforts towards understanding and generating narratives as we perceive the world and make sense of it by building a coherent sequence of units of understanding. However, most of the early systems had a modular approach to generating text. Each module handles a single narrative property of the text among content, structure, and surface-form realization. With the advent of neural techniques, the more recent modeling choice of generating text uses an end-to-end paradigm. These neural text generation models like (Radford et al., 2019; Zhang et al., 2017; Fedus et al., 2018a; Lu et al., 2018a) are becoming more and more potent in generating near-human quality text. They are very good at generating fiction but sometimes can be self-contradictory, disorderly, and inconsistent. We as a community are yet to fill in the missing gaps to make them more embedded in the context of the narrative properties from multiple views to improve the

naturalness of the text. To this end, prior and ongoing efforts towards generating a compelling narrative are concentrated on bringing synergy between the early methods on improving individual models in an end-to-end paradigm such as, including plot or event (Berman, 1988; Ammanabrolu et al., 2019b), characters (Cavazza et al., 2002; Fan et al., 2019a) development along with a dramatic arc and suspense.

However, optimizing the objective for each narrative property individually additionally demands rich annotations of these corresponding properties for each time step of the generation. With the ever-increasing amount of data, gathering annotations for narrative properties sometimes require domain expertise (for instance, content in medical documents, radiology images etc.). Therefore, it is essential to keep in mind an array of granularities in annotations to choose from for anchoring. Hence, this framework helps us visualize the spectrum of supervision needed at every time step in terms of providing anchor units between supervised and unsupervised learning paradigms. N-Local anchoring requires annotations for each utterance in a narrative, guiding what or how to say it in the context of the rest of the narrative. As each generated unit of text is rigidly and tightly bound to the anchor points, it demands such dense annotations. N-Local anchoring is more towards supervised learning with fine-grained annotations. On the other hand, N-Global is more towards semi or unsupervised learning with respect to annotations for anchor units. A similar extrapolation to reinforcement learning for N-Local anchoring is through reward shaping by providing dense intermediate rewards. For N-Global anchoring, this is done through sparse rewards, for instance, at the end of the sequence. Deriving the anchors does not need granular annotations from the training data as they are implicitly identified at the sequence level. In this thesis, we will utilize this global guidance in the context of driving or framing a narrative.

Task Based Motivations: This framework also supports the prerequisites of specific task-based stipulations. While certain tasks like storytelling can be anchored in both ways, there are still specific tasks suitable for one kind of anchoring more. For instance, N-Local anchoring is more suitable for dense plot units in stories and goal-oriented dialogs, which are usually more constrained and to the point. In contrast, N-Global anchoring is more suitable for less constrained stories that are conditioned just on a theme and chit-chat dialogs where there is room to wiggle for more diversity. This contrast is because: (i) N-Local anchoring enforces strict constraints, which is comparatively less in the case of N-Global anchoring, and (ii) N-Global anchoring has more diversity which decreases as we move towards N-Local anchoring. The task at hand dictates the degrees of freedom available for controlling these properties, and an appropriate level between N-Global and N-Local anchoring is to be applied in such cases.

While these motivations stipulate the reasoning behind the choice of anchoring, it is also important to combine this with what makes them effective in a broad typology of narratives.

2.1.2 Typology of Narratives

Before delving into how the anchors are modeled, it is important to understand the difference in narrative patterns that the anchors can bring forth. I present a high-level view of some of the different types of narratives and their characterizations regarding the narrative properties

discussed earlier. Based on the intent or the communicative goal, narratives can be broadly categorized into literary, factual, and persuasive. This categorization is not exhaustive by any means but is depicted to serve the characterization of distinction among them in the narrative properties.

- **Literary Narratives:** These types of narratives present a story. They can sometimes be voiced as external to the narrative or part of the narrative, spreading across the spectrum from a fictional story about imaginative events to a real story constituting actual original happenings.
 - *Content:* Since one of the primary objectives of these narratives is conveying a story, the content mainly revolves around the characters, places, etc.,
 - *Structure:* A coherent presentation of a story often sets the scene along with place and era. It then transcends into the characterization of the actors in terms of physical appearance and behavior. This is often followed by a sequence of obstacles and how the characters overcome them.
 - *Surface-form Realization:* Stories are creative forms of text. Hence, this form of narratives comprises key differences in the surface forms. They are metaphoric, sarcastic, and humorous. In addition, the choice of the vocabulary of each character reflects the personality of that particular character.
- **Factual Narratives:** These narratives typically present an explanation of an accord of events or describe a procedure or report factual information. Thus, the fidelity of the text is a critical property instead of being imaginative like in the earlier category.
 - *Content:* These narratives are an accounted description of the entities, events, and actions that do not deviate from the truth value. For instance, in the case of procedures, the narrative describes the ingredients or actions necessary for the successful execution of the instructions.
 - *Structure:* These factual descriptions often begin with a central introductory sentence, address different aspects or attributes of the facts. In the case of goal-oriented texts or procedures, the narrative usually starts with a statement of the goal followed by a list of materials needed to accomplish the goal. Then it talks about the step-wise description until the narrative arc attains an end goal of the final state in the instructions.
 - *Surface Realization:* The reader of the narrative is usually referred to in each step. This is either done by addressing in a very generic manner using pronouns (for example, 'you') and sometimes even omitting them. Sentences are imperative in style, indicating that the writer is giving the instructions. Intermittently, there are linking words to indicate a sequence of steps.
- **Persuasive Narratives:** These narratives attempt to change the belief or inclination of the reader or listener to a different and often to a predetermined stance. They manifest in the forms of a debate, an argument or an advertisement.
 - *Content:* In order to persuade and alter the opinion or decision through a narrative, it is crucial to gather background information. Moreover, the content should also include evidence based on reasons and examples to prove the case in point.

- *Structure*: This form of narrative usually begins with the description of the current position or stance. Then it leads its way into the presentation of alternate points of view followed by presenting logical reasons and evidence to the favored case.
- *Surface Realization*: The terms are chosen to pose convincing language. This presents a confident style, such as using words like ‘will be better’ rather than ‘might be’.

Broadly, we deal with two kinds of text narratives in this thesis: literary and factual. In particular, for the literary domain, we deal with visual stories. Storytelling through pictures has been dated back to prehistoric times. Around 30,000 years ago, paintings of animal herds like bison, rhinos and gazelles were made in a cave in Southern France. However, these were not merely paintings; they were also stories about the heroic adventures of humans. Since then, visual storytelling has evolved from paintings to photography to motion pictures to video games. However, with respect to its timeline, neural generative storytelling has gained traction only recently. Similarly, for the factual domain, I present work on procedural or instructional text (under the ‘*how-to*’ umbrella). This includes cooking recipes and navigational instructions. Along the same vein, query-oriented summaries also provide factual descriptions based on a question.

These motivations bring us to our grand goal which is discussed here.

Goal: The overarching aim of this work is to computationally simulate “**Multiview Narrative Generation**”. Let us break down what these individual terms are in understanding this goal discretely in a bottom-up fashion.

- *Generation* is the process of producing (natural language) output.
- A *Narrative* is a chronicle of a set of connected events. So far, this means that ‘*narrative intelligence*’ is the ability to craft a multi-sentence textual composition and present it perceptually. It includes knowing what to talk about and how to talk about it.
- *Multiview* is characterized by heterogeneous contexts of information. Information processing by humans is essentially multimodal in nature and situated context can be derived from multiple modalities such as vision and language.

This builds up our definition of multiview narrative generation.

Though our goal for this thesis does not encompass generating full-fledged novels and thesis documents, I present techniques towards addressing these goals in a step by step manner contributing my two cents towards this grand goal. In this attempt, I make simplified assumptions to tackle the problems in concrete ways which are discussed next.

Simplified Conjectures: The natural question that arises at this point is with respect to decoupling the differences in the narrative properties. *What is the difference between structure and content?* Both content and structure work in synergy to bring forth the semantics of the

text. With respect to this work, I see content as the individual components and structure as the arrangement of those components. For instance, in the case of a sentence: tokens represent the content and a parse tree represents the structure. In the case of an image of a cat: pixels pertaining to the eyes, ears and whiskers make up the content; and the placement of these parts appropriately comes from the structure.

Although we recognize the significance of extracting the anchors, the scope of this thesis lies in utilizing the extracted anchors in the end goal of generation. Towards this, we use readily available techniques to extract the anchors and represent them in various forms to incorporate them in the generation process. The necessary background on text generation modeling along with categorization of some of the prior work into N-Local and N-Global anchoring is presented in the next section.

2.2 Task Agnostic Text Generation Techniques

Neural text generation metamorphosed into several critical natural language applications ranging from text completion to free-form narrative generation. In order to progress research in text generation, it is critical to absorb the existing research works and position ourselves in this massively growing field. Specifically, this section surveys the fundamental components of modeling approaches relaying task agnostic impacts across various generation tasks such as storytelling, summarization, translation etc., In this context, we present an abstraction of the imperative techniques with respect to learning paradigms, pretraining, modeling approaches, decoding, and the key challenges outstanding in the field in each of them. Thereby, we deliver a one-stop destination for researchers in the field to facilitate a perspective on where to situate their work and how it impacts other closely related generation tasks.

One of the fields witnessing a steep growth is text generation, which is the task of producing a written or spoken narrative from structured or unstructured data. This field navigated through a variety of techniques and challenges from using template-based systems, modeling discourse structures, statistical methods to more recent autoregressive deep nets, transformers etc., With this rapid transformation, it is critical to retrospect and position ourselves to foresee the upcoming task-agnostic challenges to impact the entire field. *The primary goal of this section is to assist the readers to position their work in this vast maze of text generation to identify new challenges and secondarily present a compact survey of the field in the context of task-agnostic challenges.*

Before diving into the tasks-agnostic techniques, table 2.1 presents the three main paradigms of tasks in generating text based on the schema of input and output. These categories are presented for the sake of completeness of the topic at a high level but we do not go into their details in this section. These several tasks deserve undivided attention and accordingly, they have been heavily surveyed in the recent past. For instance, independent and exclusive surveys are periodically conducted on summarization (Lin and Ng, 2019; Allahyari et al., 2017; Nenkova and McKeown, 2012; Tas and Kiyani), knowledge to text generation (Gardent et al., 2017; Koncel-Kedziorski et al., 2019), machine translation (Chu and Wang, 2018; Dabre et al., 2019; Chand, 2016; Slocum, 1985), dialog response generation (Liu et al., 2016b; Montenegro

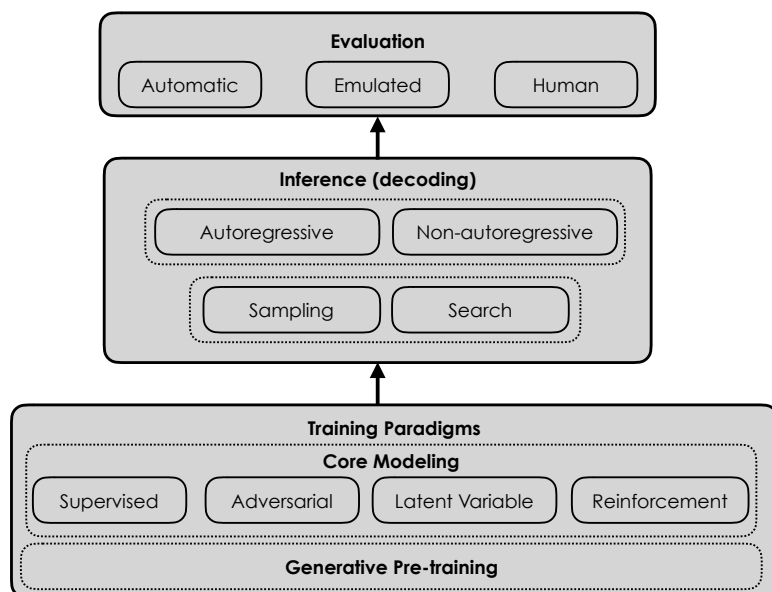


FIGURE 2.4: Outlining the components of neural text generation discussed through the section.

et al., 2019; Ramesh et al., 2017; Chen et al., 2017), storytelling, narrative generation (Tong et al., 2018; Togelius et al., 2011), image captioning (Hossain et al., 2018) etc., to dig deeper into task specific approaches that are foundational as well as the bleeding edge of research. In addition, there have been several studies conducted on surveying text generation. Perera and Nand (2017) present a detailed overview of information theory based approaches. Iqbal and Qureshi (2020) primarily focus on core modeling approaches, Gatt and Kraemer (2018a) elaborated on tasks such as captioning, style transfer etc., with a primary focus on data-to-text tasks. Controllability aspect is explored by Prabhumoye et al. (2020). Lu et al. (2018b) perform an empirical study on the core more modeling approaches only. While they are extremely necessary, the focus on techniques that are beneficial to other related tasks are often overlooked. This section focuses on these task agnostic components to improve the ensemble of tasks in neural text generation.

Figure 2.4 presents the components that are important to study in neural text generation which are elaborated in this section. Throughout the section, we identify and highlight (in italics) the challenges in the field in the context of existing work.

2.2.1 Training Paradigms

2.2.1.1 Generative Pre-training

Recent couple of years have seen a major surge in interest for pre-training techniques. UniLM (UNified pre-trained Language Model, (Dong et al., 2019a)) is proposed as a pre-training mechanism for both natural language understanding and natural language generation tasks. Fundamentally, the previously widely used ELMO (Peters et al., 2018) constitutes a language model that is left to right and right to left. While GPT (Radford et al.) has an autoregressive left to right language model, BERT (Devlin et al., 2019a) has a bidirectional language model. UniLM

Generation Paradigm	Task	Input	Output
Text-to-Text	Dialog	Conversation History	Next Response
	Machine Translation	Source Language	Target Language
	Style Transfer	Style 1 Text	Style 2 Text
	Summarization	Single/Multiple Documents	Summary
Data-to-Text	Image Captioning	Image	Descriptive Text
	Visual Storytelling	Images	Descriptive Text
	Speech Recognition	Audio	Text
	Table to Text	Table	Text
	Knowledge Bases to Text	Knowledge Bases	Text
None-to-Text	Language Modeling	Null	Sequence of Text

TABLE 2.1: Paradigms of Tasks in Text Generation (not detailed in this chapter). Note: To be compact, we include ‘Knowledge-to-text’ paradigm within ‘Data-to-text’.

is optimized jointly for all of the above objectives along with an additional new seq2seq LM which is bidirectional encoding followed by unidirectional decoding. Depending on the use case, UniLM can be adopted to use Unidirectional LM, Bidirectional LM, and Seq2seq LM. With a similar goal in mind, MASS (Song et al., 2019) modified masking patterns in input to achieve this. BERT and XLNet (Yang et al., 2019d) pre-train an encoder and GPT pretrains a decoder. This is a framework introduced to pretrain encoder-attention-decoder together. Encoder masks a sequence of length k and the decoder predicts the same sequence of length k and every other token is masked. While the idea of jointly training the encoder-attention-decoder remains the same as in UniLM, the interesting contribution here is the way masking is utilized to bring out the following advantages. (i) The tokens masked in decoder are the tokens that are not masked in encoder. (ii) Encoder supports decoder by extracting useful information from the masked fragments improving the NLU capabilities. (iii) Since a sequence of length k is decoded consecutively, NLG capability is improved as well. Note that when k is 1, the model is closer to BERT which is biased to an encoder and when k is the length of sentence, the model is closer to GPT which is biased to decoder. Similar to UniLM, BART (Lewis et al., 2019) has a bidirectional encoder and an autoregressive decoder. The underlying model is standard transformer (Vaswani et al., 2017a) based neural MT framework. The main difference between BART and MASS is that the tokens masked here are not necessarily consecutive. The main idea is to corrupt text with arbitrary noise such as token masking, token deletion, token infilling, sentence permutation, and document rotation and reconstruct the original text. Following this, Raffel et al. (2019) proposed T5 as a unifying framework that ties all NLP problems as text generation tasks with a text-in and text-out paradigm. Recently, Dathathri et al. (2020) introduced plug and play language models capable of efficiently training fewer parameters to control a huge underlying pretrained model. Finetuning these vast models for generative tasks has been studied in style transformers (Sudhakar et al., 2019) and conversational agents (Dinan et al., 2019).

Challenges: This new era of very powerful language models opened up a whole new set of challenges. *How can they be effectively used as knowledge sources (Lewis et al., 2020) ? How to mitigate the inherent societal biases from models learnt at this scale (Shwartz et al., 2020a) ? How to learn social norms from vast amounts of pretrained models (Forbes et al., 2020) ? How to ensure coherence in long form generation specific to a domain (Tan et al., 2020) ?*

2.2.2 Core Modeling

The base architecture constitutes of an encoder-decoder with an optional attention mechanism.

Supervised Learning: Most generation approaches in this setting use maximum likelihood objective for training sequence generation with a sequential multi-label cross entropy. However, there is an inherent inconsistency in exposure to ground truth text between training and inference stages when using teacher forcing during training, leading to exposure bias (Ranzato et al., 2016). This problem becomes severe with the increasing length of the output. A solution to address this issue is scheduled sampling (Bengio et al., 2015) which mixes teacher forced embeddings and model predictions from the previous time step. *This problem of text degeneration still remains a pressing issue*, which is revisited in §2.2.4.

Reinforcement Learning: The main issue with the supervised learning approach for text generation is the mismatch between the maximum likelihood objective and metrics for text quality. Reinforcement learning addresses this mismatch by directly optimizing end metrics which could be non-differentiable. Typically, policy gradient algorithms are used to optimize for BLEU score directly via Reinforce. However, *computing these metrics before every update is not computationally efficient to incorporate in the training procedure*. Another problem is the *inherent inefficiency of the metric itself used for reward* i.e BLEU is not the best measure to evaluate text quality. Sometimes these rewards are learnt adversarially. In practice, usually, the policy network is usually pre-trained with maximum likelihood objective before optimizing for BLEU score.

Latent Variable Modeling: These models took a pretty steep curve from variational seq2seq models (Bowman et al., 2016) to conditional VAEs, (Shen et al., 2017b). These latent variable models have been explored to generate controllable text generation based on topic (Wang et al., 2019e), structure (Chen et al., 2019a), persona (Zhao et al., 2017) (Wu et al., 2020) etc.,

Adversarial Learning: The third paradigm is adversarial learning comprising of competing objectives. The mismatch in training and inference stages is addressed using Professor Forcing (Lamb et al., 2016) with adversarial domain adaptation to bring the behavior of the training and sampling close to each other. Generative Adversarial Networks (GAN) also gained popularity with respect to this in the recent times. The core idea is that the gradient of the discriminator guides how to alter the generated data and by what margin in order to make it more realistic. There are several variants adopted to address specific problems such as SeqGAN to assess partially generated sequence (Yu et al., 2017), MaskGAN to improve sample quality using text filling (Fedus et al., 2018a) and LeakGAN to model long term dependencies by leaking discriminator information to generator (Guo et al., 2018). The three main challenges researched in this area are:

- *Discrete Sampling:* The sampling step selecting argmax in language is non-differentiable. This is addressed by replacing it with a continuous approximation by adding Gumbel noise which is negative log of negative log of a sample from uniform distribution, also known as Gumbel Softmax.
- *Mode Collapse:* GANs typically face the issue of sampling from specific tokens to cheat discriminator, known as mode collapse. In this way, only a subspace of target distribution is

learnt by the generator. DP-GAN addresses this using an explicit diversity promoting reward (Xu et al., 2018b).

- *Power dynamics between Generator and Discriminator during training:* Another problem arises when the discriminator is trained faster than the generator and overpowers it. This phenomenon in training is observed frequently, and the gradient from discriminator vanishes leading to no real update to generator.

2.2.3 Decoding Strategies

The natural next step after pre-training and training is decoding. The distinguishing characteristic of generation is the absence of one-to-one correspondence between time steps of input and the output, thereby introducing a crucial component which is decoding. Primarily, they can be categorized as (i) autoregressive and (ii) non-autoregressive.

Autoregressive decoding: Traditional models with this strategy correspond well to the true distributions of words. This mainly comes from respecting the conditional dependence property from left to right. The autoregressive techniques can be further viewed as sampling and search techniques. One of the issues of this strategy is *throttling transformer-based models that fall short in replicating their training advantages as training can be non-sequential* and inference holds to be sequential with autoregressive decoding.

Non-autoregressive decoding: This line of work primarily addresses two problems that are associated with autoregressive decoding. First, by definition, there is a conditional independence property that holds. This leads to the *multimodality problem, where each time step considers different variants with respect to the entire sequence* and these conditions compete with each other. Second, the main advantage is the reduction in latency during real-time generation. Guo et al. (2020) addressed this problem in the context of neural machine translation using transformers by copying each of the source inputs to the decoder either uniformly or repeatedly based on their fertility counts. These fertilities are predicted using a dedicated neural network to reduce the unsupervised problem to a supervised one and thereby enabling it to be used as a latent variable. These invariable replications based on fertilities may lead to *repetition or duplication of words*. Closely followed by this, van den Oord et al. (2018) took a different approach by introducing probability density distillation by modifying a convolutional neural network using a pre-trained teacher network to score a student network attempting to minimize the KL divergence between the teacher network and itself. Both these works set the trend of using latent variables to capture the interdependence between different time steps in the decoder. Following this work, Lee et al. (2018) use iterative refinement by denoising the latent variables at each of the refinement steps. This idea of iterative decoding inspired ways to more avenues by combining the benefits of cloze style mask prediction objectives from Bert (Devlin et al., 2019a). Some of them include insertion based techniques (Gu et al., 2019a), repeated masking and regenerating (Ghazvininejad et al., 2019) and providing model predictions to the input (Ghazvininejad et al., 2020). Wang et al. (2019e) proposed an alternative approach to address repetition (observed in Guo et al. (2020)) and completeness using regularization terms for each. Repetition is handled by regularizing similarity between consecutive words. Completeness is addressed by enabling the reconstruction of the source sentence from hidden

states of the decoder. Concurrently, [Guo et al. \(2019a\)](#) also address these issues by improving the inputs to decoder using additional phrase table information and sentence level alignment.

Sampling and Search Techniques:

1. Random Sampling: The words are sampled randomly based on the probability from the entire distribution without pruning any of the mass.

2. Greedy Decoding: This technique simply boils down to selecting argmax of the probability distribution. As you keep selecting argmax everywhere, the problem is that it limits the diversity of generation. Temperature scaling helps to adjust the spectrum between flat and peaky distributions to generate more diverse or safe responses respectively. This is alleviated by the following techniques and beam search. This is also worked out for discrete settings using gumbel-greedy decoding ([Gu et al., 2018a](#)). Variants of this were also studied by [Zarrieß and Schlangen \(2018\)](#).

3. Beam Search: Beam search introduces a course correction mechanism in approximation of the argmax by selecting a beam size number of beams at each time step. It has been relatively well studied in task agnostic objectives ([Wang et al., 2014](#)) for instance, including social media text ([Wang and Ng, 2013](#)), error correction ([Dahlmeier and Ng, 2012](#)). *Small beam sizes may lead to ungrammatical sentences*, they get more grammatical with increasing beam size. Similarly, small beam sizes may be less relevant with respect to content but get more *generic with increasing beam size*. Prominent variants within beam search are:

(a) Noisy Parallel Approximate Decoding: This method ([Cho, 2016](#)) introduces some noise in each hidden state to non-deterministically make it slightly deviate from argmax.

(b) Beam Blocking: Repetition is one of the problems we see in NLG and this technique ([Paulus et al., 2018](#)) combats this problem by blocking the repeated n-grams. It essentially adjusts the probability of any repeated n-gram to 0.

(c) Iterative Beam Search: In order to search a more diverse search space, another technique ([Kulikov et al., 2019](#)) was introduced to iteratively perform beam search several times. And for each current time step, we avoid all of the partial hypotheses encountered until that time step in the previous iterations based on soft or hard decisions on how to include or exclude these beams.

(d) Diverse Beam Search: One problem with beam search is that most times the decoded sequence still tends to come from a few highly significant beams thereby suppressing diversity. The moderation by ([Vijayakumar et al., 2016](#)) adds a diversity penalty computed (for example using Hamming distance) between the current hypothesis and the hypotheses in the groups to readjust the scores for predicting the next word.

(e) Clustered Beam Search: The goal is to prune unnecessary beams. At each time step, [Tam \(2020\)](#) get the top candidates and embed them by using averaged Glove representations which are clustered using k-means to pick from each cluster.

(f) Clustering Post Decoding: This technique ([Kriz et al., 2019](#)) clusters after decoding as opposed to modifying the decoding step. Sentence representations from any of the diversity

promoting beam search variants are obtained. These are then clustered and the sentence with high log-likelihood is selected from the cluster.

4. Top-k sampling: This technique by Fan et al. (2018) samples from the k most probable candidates from the output distribution. This means that we are confining the model to select from a truncated probability mass. If k is the size of vocabulary, then it is random sampling and if k is 1 then it is greedy decoding. High valued k results in dicey words but are non-monotonous and low valued k results in safe outputs which are monotonous. The problem however is that k is limited to the same value in all scenarios.

5. Top-p sampling: The aforementioned problem of a fixed value of k is addressed by top-p sampling. This is also known as nucleus sampling (Holtzman et al., 2020), which instead of getting rid of the unspecified probability mass in top-k sampling, importance is shifted to the amount of probability mass preserved. This addresses scenarios where there could be a broader set of reasonable options and sometimes a narrow set of options. It is achieved by selecting a dynamic k number of words from a cumulative probability distribution until a threshold probability value is attained.

2.2.4 Key Challenges

For each of the challenges, this section provides a list of solutions. The pitfalls of these solutions are also described there by encouraging research to address these key challenges.

1. Content Selection: Certain tasks demand copying over the details in the input such as rare proper nouns for instance in news articles etc., This is especially needed in tasks like summarization which can demand a combination of extractive and abstractive techniques.

- **Copy Mechanism:** Copy mechanism can take various forms such as pointing to unknown words (Gulcehre et al., 2016) based on attention (See et al., 2017) or a joint or a conditional copy mechanism (Gu et al., 2016; Puduppully et al., 2019). It may be based on attention that copies segments from input into the output. *The challenge in this technique is to make sure that this combination of being extractive and abstractive does not boil down to a purely extractive system.*

- **Attention mechanism:** This is actively used for subselecting content in tasks such as summarization (Chopra et al., 2016). Recent work has demonstrated instances of *attention not explaining the output* (Jain and Wallace, 2019; Latacinnik and Berant, 2020).

- **Hierarchical Modeling:** This technique maintains a global account of the content. This is often modeled using hierarchical techniques or dual-stage models (Martin et al., 2018; Xu et al., 2018a; Gehrmann et al., 2018) where the first stage pre-selects relevant keywords for generation in the following stage. *Such models possibly take a hit on fluency while connecting dots between selected content and generation.* This means that Rouge-1 can be good because the right words are extracted but Rouge-2 may decrease as it affects the fluency.

- **Memory Modules:** Zhou et al. (2018b) and Clark et al. (2018a) explored memory modules in inducing emotion and entity representations from external memory respectively. *An outstanding challenge still remains in exploring the best ways to encode this external memory.*

2. Repetition: Holtzman et al. (2020) demonstrate that the objective of maximum likelihood renders high log likelihood for the words that have been generated, leading to repetition. This problem amplifies with increasing sequence length and transformer-based models.

- **Beam blocking:** Blocking beams containing previously generated n-grams from subsequent generation combats repetition and encourages diversity (Klein et al., 2017; Paulus et al., 2018) etc., *Selecting the number of beams is often a problem since it is natural for a function word to repeat more often.* Massarelli et al. (2019) extensively studied the variants of n-gram blocking by applying delays in beam search.

- **Unlikelihood objective:** Welleck et al. (2020) argue that there is a fundamental flaw in the objective of maximum likelihood. The main idea is to decrease the probability of unlikely or negative candidates. The negative candidates are selected from the previous contexts either at token or at sequence levels which are essentially n-grams. *Selecting negative contexts is tricky and needs to be beyond selection of simple n-gram sequences that occurred previously.*

- **Coverage penalty:** This discourages the attention mechanism to attend the same word repeatedly (See et al., 2017) by assigning coverage penalty of the attention probability mass on that source time step for each decoded time step.

3. Coherence: This is a critical property of text to factor in for multi-sentence or long-form generation, that not only takes into account the appropriate content but also the structure of the narration.

- **Static and Dynamic Planning:** This addresses coherence in terms of layout or structural organization of the text (Yao et al., 2019). A schema of static or dynamic plans is used to form an abstract flow of the text from which the actual text is realized. *However, underlying language models are capable of taking over, leading to hallucinations and thereby compromising the fidelity of text.*

4. Length of Decoding: One factor that distinguishes generation from the rest of the seq2seq family of tasks is the variability in the length of the generated output. The main problem here is that as the length of the sequence increases, the sum of the log probability scores decreases. This means that *models prefer shorter hypotheses*. Some solutions to combat this problem are the following.

- **Length Normalization or Penalty:** The generated output is scored by normalizing or dividing with length. Wu et al. (2016b) explore a different variation of the normalization constant.

- **Probability boosting:** This technique multiplies the probability with a fixed constant at every time step. This alleviates the diminishing score problem.

- **Length based bias:** Incorporate bias in the model based on empirical relations on lengths in source and target sentences in the training data.

5. Sequence level scoring: Instead of modifying the decoding, this strategy performs sequence level scoring from multiple texts decoded.

• **Reranking:** Another mechanism is to sample several full sequences and rerank them based on generic scores such as perplexity or BLEU or task specific requirements varying from factual correctness (Goyal and Durrett, 2020), coherence (Lu et al., 2020), style (Holtzman et al., 2018) etc., *The properties of text in the end goal are decoupled from the generation process* (a soft conditioning of reranking on generation helps improve the generation as well).

5. Optimization Objective: Similar to the observation earlier in §2.2.1, there is an inherent mismatch in between the objective function which is maximum likelihood and the end metrics which are BLEU, Rouge etc;

• **Reinforcement Learning:** A common solution for this problem is using reinforcement learning to optimize end metrics such as Rouge. Often, a combination of MLE and RL objectives are used (Hu et al., 2020a; Wang et al., 2018) to optimize BLEU (Wu et al., 2016a), ROUGE (Paulus et al., 2018), CIDEr (Rennie et al., 2017), SPIDEr (Liu et al., 2017a). *An existing open challenge is to understand how to make the models robust by making them learn the task rather than gaming for the reward.* The rewards can also be learnt adversarially during the training (Li et al., 2017). *However, this is still a problem since these end metrics do not directly correlate to human judgments.*

• **Factorizing Softmax:** Choi et al. (2020) recently proposed a method to factorize softmax by learning to predict both the frequency class and the token itself during training by factorizing the probabilities. *This model is observed to repeat the same rare token across several sentences.*

• **Maximum Mutual Information:** The idea is to incorporate pairwise information of source and target instead of only one direction which is usually target given source (Li et al., 2016a). The target probability is subtracted from target given source probability to diminish the probability of generic sentences. *The model optimized with MMI can sometimes generate ungrammatical sentences.*

• **Distinguishability:** Hallucinations in abstractive generation are unwanted byproducts of optimizing log loss. To combat this, several researchers explored optimizing for minimized distinguishability with human generated text (Hashimoto et al., 2019; Theis et al., 2016). Following similar path, Kang and Hashimoto (2020) proposed truncating loss to get rid of unwanted samples.

5. Speed: Practical applications call for a crucial research direction of generating text in real-time in addition to chasing state-of-the-art results. Model compression plays a crucial part in demonstrating an increase in the speed of generation. Cheng et al. (2017) exhaustively surveyed the different techniques to perform model compression. While there are techniques in the hardware side, there are certain modeling approaches that can handle this problem as well (Gonzalvo et al., 2016). Most of this work is studied in the context of real-time interpretation of speech (Fügen et al., 2007; Yarmohammadi et al., 2013; Grissom II et al., 2014). Recently, Deng and Rush (2020) proposed a cascaded decoding approach introducing Markov Transformers demonstrating high speed and accuracy.

- *Quantization*: Quantizing (Roy et al., 2018; Gray, 1984) the weights, i.e., sharing the same weight when they belong to a bin, proved helpful in improving the speed. This also facilitates the computations of gradients only once per bin.
- *Distillation*: It can be performed with a teacher and a smaller student network that tries to replicate the performance of the teacher with fewer parameters (Chen et al., 2019c, 2020c).
- *Pruning*: This technique thresholds and prunes all the connections that have weights lesser than the predetermined threshold and then we can retrain the network in order to adjust the weights of the remaining connections.
- *Real time*: Gu et al. (2017) trained an agent that learns to decide between the actions of reading by discarding a candidate or writing by accepting a candidate. The policy network is optimized with a combination of quality evaluated with BLEU and delay evaluated by the number of consecutive words in the reading stage which increases wait time.
- *Caching*: Another trick is to cache some of the previous computations to avoid repetition.

2.2.5 Evaluation

Similar to other generative modeling, text generation also faces crucial challenges in evaluation (Reiter and Belz, 2009; Reiter, 2018). van der Lee et al. (2019) present some of the best practices of evaluating automatically generated text. The main hindrance to standardize or evaluate NLG like other standard tasks is that it is often a sub-component of other tasks. Celikyilmaz et al. (2020) present a more comprehensive survey of evaluation metrics for text generation.

Desiderata of Text: It is crucial to define the factors contributing to the quality of good text. Some of the factors include relevant content, appropriate structure in terms of coherence, and suitable surface forms. In addition, fluency, grammaticality, believability, and novelty in some scenarios are crucial factors.

Intrinsic and Extrinsic: Evaluation in subjective scopes such as text generation can be performed intrinsically or extrinsically. Intrinsic evaluation is performed internally with respect to the generation itself and extrinsic evaluation is typically performed on the metric used to evaluate a downstream task in which this generation is used. The quality can also be judged using automatic metrics and human evaluation.

(a) Automatic Metrics: These metrics can be classified into the following categories:

- **Word overlap based metrics:** These are based on the extent of word overlap, which means that they capture replication of words. The problem with such measures is that they do not focus on semantics but rather just the surface form of words alone. This includes precision for n-grams (BLEU (Papineni et al., 2002a)), self-BLEU (Zhu et al., 2018), improved weighting for rare n-grams (NIST (Doddington, 2002)), recall for n-grams (ROUGE (Lin and Hovy, 2002)), F1 equivalent of n-grams (METEOR (Banerjee and Lavie, 2005a)), tf-idf based cosine similarity for n-grams (CiDER (Vedantam et al., 2015)). In extension to this, we also have specific metrics to evaluate content selection by measuring summarization content units using PYRAMID (Nenkova and Passonneau, 2004) and parsed scene graphs with objects and relations using

SPICE (Anderson et al., 2016). Stanojevic and Sima'an (2014) proposed BEER to address this as a ranking problem with character n-grams along with words.

• **Language Model based metrics:** This includes perplexity (Brown et al., 1992). Such metrics are good in commenting about the language model itself. It sort of gives the average number of choices each random variable has. However, it does not directly evaluate the generation itself, for instance, a decrease in perplexity does not imply a decrease in the word error rate. *The problem remains that this metric intrinsically conveys if the LM is good enough to select the right next word for that corpus but not the actual quality of the generated text.* The human likeness is also measured by training a model to discriminate between human and machine-generated text such as an automatic Turing test (Lowe et al., 2017; Cui et al., 2018; Hashimoto et al., 2019).

• **Embedding based metrics:** This has the advantage of being able to capture distributed semantics compared to word overlap metrics and language model based metrics. MEANT 2.0 (Lo, 2017) and YISI-1 (Lo et al., 2018) computes structural similarity with shallow semantic parses being definitely and discretionarily used respectively along with word embeddings. Recently, contextulized embeddings have been extensively used to capture this, such as BertScore (Zhang et al., 2020b) and BLEURT (Sellam et al., 2020). Metrics based on a combination of different embeddings are also proposed (Shimanaka et al., 2018; Ma et al., 2017). *However the problem of not correlating to human judgements still persists.*

(b) Emulated Automatic Metrics: These metrics check for the intended behavior in generation based on the specific sub-problem being addressed. For instance, diversity can be evaluated by computing corpus based distributions on number of distinct entities (Fan et al., 2019b; Dong et al., 2019b; Clark et al., 2018a) and so on. Recent approaches worked on identifying factual inconsistencies with a QA model using QAGS (Wang et al., 2020a), answering cloze style questions using SummaQA (Scialom et al., 2019), performance on a language understanding task using BLANC (Vasilyev et al., 2020), adhering to pre-defined commonsense conditions (Gabriel et al., 2020).

(c) Human Evaluation: There are broadly two mechanisms in conducting subjective evaluations which is a challenging component of text generation. The first is preference testing and the second is scoring. Studies have shown that preference-based testing is prone to high variance compared to absolute scoring. Here are some important points to keep in mind during conducting human evaluation. There are several problems with human evaluations. They are expensive, have no universally agreed upon guidelines for setup, are difficult to ensure quality control, have varying scores based on scales (binary vs continuous), are difficult to replicate, presenting the task in an unambiguous way. In order to measure more reliably, we need to collect multiple scores and compute inter-annotator agreement with Cohen's, Krippendorff's coefficients etc., Having critically discussed human evaluation, this is still really the best we got. It is absolutely crucial to perform human evaluation in most tasks. So, the aforementioned problems need to be taken merely as cautions to develop rational and systematic testing conditions. Comparisons between automatic and human evaluation metrics (Belz and Reiter, 2006) are actively studied in order to bring human evaluations closer to automatic metrics.

2.3 Modeling Approaches for Generation and Anchoring

Prior methods in generation primarily had a modular approach dealing with each of the narrative properties that heavily rely on designed rules and simple surface form similarity or probability modeling. With the advent of neural models, text generation systems have transcended into an encoder-(attention)-decoder framework. Some of the popularly used techniques are multi-task learning, adversarial generation, external reward such as reinforcement learning, structured loss, scalar manipulation, post-editing, attention, latent variable. I will briefly give an overview of these techniques to facilitate a better understanding of which techniques are used to improve which properties and in which anchoring techniques in our work.

- **Multi-task Learning:**

This is a technique to perform inductive transfer by the means of an auxiliary task in addition to our primary task. As pointed out by [Ruder \(2017\)](#), multitask learning acts implicit data augmentation and eavesdropping. We are going to utilize these properties in learning the languages of the words in code-switched settings in addition to decoding the words itself. Leveraging a related task aids in transferring generalized information to another closely related task. Multitask learning not only acts as a regularizer in this aspect but also learning robust representations while training for the auxiliary task.

The objective for a single task (\mathcal{T}) learning as in the case of regular supervised learning setting is:

$$\min_{\theta} \mathcal{L}(\theta, \mathcal{T}) \quad (2.1)$$

Extending this to a multitask learning objective makes it:

$$\min_{\theta} \sum_{i=1}^{\mathcal{T}} \mathcal{L}_i(\theta, \mathcal{T}) \quad (2.2)$$

Conditioning on the task itself can be formalized as either multiplicative gating, addition, concatenation or using multiple heads for different tasks. However, a common challenge that comes to play with the architecture is negative transfer. This happens when the tasks are not sufficiently similar leading to the independent network performing better than multitasking. This can be overcome by sharing parameters softly and weighing them with a small constant (α).

$$\min_{\theta_{sh}, \theta_1, \dots} \sum_{i=1}^{\mathcal{T}} \mathcal{L}_i(\{\theta, \theta_i\}, \mathcal{T}) + \alpha \sum_{t'=1}^{\mathcal{T}} \|\theta_t - \theta_{t'}\| \quad (2.3)$$

While identifying a related task is a challenge, this technique in combination with disentanglement has shown promising results to transfer style without varying the semantics of a sentence. This in specific is indexed in the proposed work for disentangling persona to facilitate persona-based visual storytelling.

- **Adversarial Training:** This introduces a difference in architecture with two competing networks. Generative Adversarial Networks, commonly known as GANs belong to this

clan and have attained immense popularity in computer vision. There are a few issues when adapting this modeling approach to the textual domain and we are going to briefly discuss them. The premise of this model is that there are two competing networks: a generator and a discriminator. The adversarial aspect of GANs also leads to the minimax game. This is because while the discriminator attempts to maximize its performance on classifying between real and fake, the generator strives to fool the discriminator i.e, minimize its performance.

Among the narrative properties that we described, this setup is comparatively commonly used in surface realization, while there is no reason why it cannot be used to improve the other properties. In this thesis, we are going to present an adversarial learning setup to generate text with surface realization forms from multiple languages. Here, the objective that we are going to optimize is:

$$\min_{\theta} \max_{\phi} = \mathbb{E}_{\mathbf{x} \sim p_{data}} [\log \mathbf{D}_{\phi}(\mathbf{x})] + \mathbb{E}_{\mathbf{z} \sim p(\mathbf{z})} [\log(1 - \mathbf{D}_{\phi}(\mathbf{G}_{\theta}(\mathbf{z})))] \quad (2.4)$$

For this we are going to first sample a minibatch of b sentences from true data: $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(b)} \sim \mathbf{D}$. Similarly, a minibatch of size b is sampled from latent space: $\mathbf{z}^{(1)}, \mathbf{z}^{(2)}, \dots, \mathbf{z}^{(b)} \sim p_z$.

First, a gradient descent step on generator parameters θ :

$$\frac{1}{b} \nabla_{\theta} \sum_{i=1}^b \log(1 - \mathbf{D}_{\phi}(\mathbf{G}_{\theta}(\mathbf{z}^{(i)}))) \quad (2.5)$$

Following this, a gradient ascent step on discriminator parameters ϕ :

$$\frac{1}{b} \nabla_{\phi} \sum_{i=1}^b [\log \mathbf{D}_{\phi}(\mathbf{x}^{(i)})] + [\log(1 - \mathbf{D}_{\phi}(\mathbf{G}_{\theta}(\mathbf{z}^{(i)})))] \quad (2.6)$$

When we are dealing with datasets from two different forms of surface realization, often these varieties are known as styles.

There are two different datasets $\mathbf{X} = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(n_x)}\}$ that belongs to style \mathbf{s}_x and another dataset $\mathbf{Y} = \{\mathbf{y}^{(1)}, \mathbf{y}^{(2)}, \dots, \mathbf{y}^{(n_y)}\}$ that belongs to style \mathbf{s}_y . The generator network typically has an encoder-decoder framework. The goal of attaining a disentangled representation is to divide each sentence \mathbf{x} from \mathbf{X} into the content representation \mathbf{c}_x and the corresponding style \mathbf{s}_x . Similar parallel components for \mathbf{Y} are \mathbf{c}_y and \mathbf{s}_y . The content vector for sentences \mathbf{x} and \mathbf{y} is attained by passing through an encoder \mathbf{E} i.e, $\mathbf{c}_x = \mathbf{E}(\mathbf{x}, \mathbf{s}_x)$ and $\mathbf{c}_y = \mathbf{E}(\mathbf{y}, \mathbf{s}_y)$. This is followed by a decoder \mathbf{F} that is aimed at generating a sentence \mathbf{x} in the other style \mathbf{s}_y . The decoder is attempting to reconstruct the original sentence in a different style and hence is similar to an auto-encoder. The objective for reconstruction is:

$$\mathcal{L}_{\mathbf{G}_{\theta}}(\theta_{\mathbf{E}}, \theta_{\mathbf{F}}) = -(\mathbb{E}_{\mathbf{x} \sim \mathbf{X}} [\log \mathcal{P}_{\mathbf{F}}(\mathbf{x} | \mathbf{c}_x, \mathbf{s}_x)] + \mathbb{E}_{\mathbf{y} \sim \mathbf{Y}} [\log \mathcal{P}_{\mathbf{F}}(\mathbf{y} | \mathbf{c}_y, \mathbf{s}_y)]) \quad (2.7)$$

Often the language models that can be trained in an unsupervised fashion are trained as discriminators. Discriminator is any classifier that discriminates between the two styles.

$$\mathcal{L}_{\mathbf{D}_\phi} = -\log \mathcal{P}_\phi(\mathbf{c}_\mathbf{x}|\mathbf{x}) \quad (2.8)$$

With the minimax game between the generator and the discriminator, the adversarial objective that we are optimizing is:

$$\min_{\mathbf{E}_\theta, \mathbf{F}_\theta} \max_{\mathbf{D}_\phi} \mathcal{L}_{\mathbf{G}_\theta} - \alpha \cdot \mathcal{L}_{\mathbf{D}_\phi} \quad (2.9)$$

- **External Reward:** A stumbling block is annotating texts with the narrative properties and the fine-grained classes in each of these properties. Curating a dataset to perform supervised learning is expensive in most cases. Moreover, another issue with the supervised learning approach is the mismatch in what we are trying to optimize vs how we measure the quality of the text. Reinforcement Learning when applied to generation directly optimizes end metrics that could be non-differentiable. In addition, by definition, our model is upper bounded by the accuracy of the annotations in our training data and learning other statistical regularities. Hence we use reinforcement learning which has the same architectural structure as that of our supervised learning network. This network that generates text from our input is our policy network. One of the approaches to train this network is by using policy gradients. Based on the generated text in the context that it is being used, a reward or a penalty is assigned.

Typically, using reinforcement learning for text generation maps the definitions as follows: *State* is the words in the input and the words generated in the output till the current time step. *Action* is the generation of the next token. *Policy* is the derivation of the probabilities over the entire vocabulary for decoding the next token. *Reward* determines the quality of the action. The training objective in this case is:

$$\sum_{t=1}^T \sum_{y_t \in V} \mathcal{P}_\theta(y_t|s_t) Q(s_t, y_t) \quad (2.10)$$

Here, $\mathcal{P}_\theta(y_t|s_t)$ is the likelihood of decoding the token y_t given the current state and $Q(s_t, y_t)$ is the reward for the sentence. The computation of this reward itself can be framed in various ways as a function of the probability of the tokens and the end goal of the task. For instance, this is addressed by [Hu et al. \(2020b\)](#) by optimizing rewards for relevance, coherence, and expressiveness separately. As we observe in the rest of the settings, the decoder or autoencoder responsible for generating text relies on differentiable training objectives like cycle consistency etc.,. These objectives are sometimes not directly correlated to our metrics such as METEOR, ROUGE etc.,.

One problem as we can see here is the credit assignment problem which determines the sparsity or granularity of the rewards. This is how this reinforcement learning paradigm takes its place in the proposed anchoring framework. Tailoring reward shaping to a specific task hinders scalability to other datasets and tasks.

- **Structured Loss:** While classification deals with loss term based on the number of misclassifications which are discrete and regression deals with continuous target, structured prediction deals with a structure or sequence of labels. The output space is finite but dealing with it as a classification problem with permutations of target sequence as classes is not efficient. This leads to problems in scoring if only a few tokens y_i are decoded incorrectly. This becomes even more complicated when it comes to a sequence of sentences. A structured loss term addresses these issues by considering the entire output as a whole. The sentences in a sequence can form a linear or a graphical structure with interdependencies from the rest of the sequence. This takes into account the distribution over the entire sequence and computes the distortion with the sequence in the ground truth.
- **Scalar Manipulation:** The latent representation of the input can be modified with simple arithmetic operations to condition on the anchors. This can be done at each time step for concatenation and addition respectively as follows

$$h_t = [h_t; a_t] \quad (2.11)$$

$$h_t = h_t \pm a_t \quad (2.12)$$

Similarly, this can also be done for the whole input together. While concatenation increases the model complexity owing to the size of the representation, other arithmetic operations such as addition and subtraction interfere with the token representation itself.

- **Post-editing:** This is a technique borrowed from traditional approaches which are more modular. As the name suggests, this approach edits the text gathered post the rest of the modules. Often this is done to replace designated tokens in tasks such as style transfer with sentiment etc., Another closely related task is the generation of simple text from text dense with complex jargon. The token level replacement is not a scalable approach and often requires manual labor to gather similar or contrasting lexical items. This also extends to reordering sentences at a discourse level.
- **Attention:** Model capacity determines a bottleneck in the flow of information. Attention is one such mechanism that stochastically weighs a segment of the information from input based on its significance to the end task. This quantifies the dependence and/or interdependence between the input units among themselves and also with the output units. Attention mechanism enables determining the focus or significance of the input words with respect to our end goal. This can be formalized in terms of the input hidden states (\mathbf{h}_s) and the hidden state at the current time step (\mathbf{h}_t) to arrive at the attention weights α_{ts} as follows.

$$\alpha_{ts} = \frac{\exp(\text{score}(\mathbf{h}_t, \mathbf{h}_s^-))}{\sum_{s'=1}^S \exp(\text{score}(\mathbf{h}_t, \mathbf{h}_{s'}^-))} \quad (2.13)$$

The context vector (\mathbf{c}_t) for that particular time step is then computed as a weighted sum of the attention weights and the input hidden states.

$$\mathbf{c}_t = \sum_s \alpha_{ts} \bar{\mathbf{h}}_s \quad (2.14)$$

Finally, the attended representation is computed and used for further part of the network.

$$\mathbf{a}_t = f(\mathbf{c}_t, \mathbf{h}_t) = \tanh(\mathbf{W}_c[\mathbf{c}_t; \mathbf{h}_t]) \quad (2.15)$$

In context, we have used hierarchical attention based model for anchoring in the appropriate content in the generation of visual stories. For instance, anchoring a story around characters calls for attending to these characters to weave into the text. In addition, attention-based encoders and decoders showed powerful results and led to transformer (Vaswani et al., 2017b) models. This tapped an unprecedented way to capture essential information in modeling approaches.

- **Latent Variable Modeling:** Contemporary techniques in text generation with latent variable models are instantiated in flow models and variational autoencoders. The latter models are also used to disentangle various key properties of the input. Disentanglement is deeply motivated characteristically by the need for interpretability by humans which is a crucial goal of artificial intelligence. Recent times have seen a growing interest in cohabitant pretraining and representation learning. We are not going into details about the importance of representation learning itself. The goal here is to find statistical regularities based on underlying interactions in the data. Factorizing the segments of representation corresponding to the content, structure, and surface forms enables efficacy in using this data and transferring the knowledge across tasks. Learning these specific properties implicitly within our modeling approaches potentially leads to overfitting to the data and hinders generalization. The explicit learning of these regularities motivates disentanglement. A generative model has to be non-repetitive and diverse which introduces using a variational autoencoder (VAE) in text generation settings. Let $p_\theta(\mathbf{x}|\mathbf{z})$ be the parameterized likelihood with the generative parameter θ . Similarly, let $q_\phi(\mathbf{z}|\mathbf{x})$ be the parameterized approximate posterior with variational parameter ϕ . This is explored in partial anchoring of our framework by Wang et al. (2019d). The key difference is that an input is mapped to a distribution instead of a fixed vector. Usually this distribution is a Gaussian parameterized with mean and variance.

$$\mathcal{L}(\theta, \phi, \mathbf{x}, \mathbf{z}) = \mathbb{E}_{q_\phi}[\log p_\theta(\mathbf{x}|\mathbf{z})] - D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p(\mathbf{z})) \quad (2.16)$$

In the right hand side of the equation, the first term represents the reconstruction loss and the second term forces the learnt distribution to be close to gaussian of known mean and variance.

The data itself comprises a variation in the narrative properties. This, for instance, can be seen through the source of data such as Twitter, topics, domains, genres of texts etc.,. While these are the generative factors in the data, the disentangled VAE is capable of inferring these distinct underlying factors. While we have disentangled dimensions or factors of the text, varying the degree of the corresponding factor smoothly navigates the tone of the text towards that dimension. Xu et al. (2019) attempted controllable text generation without any supervision by enforcing a structural constraint based on global

variations. In this way, they have examined transitions between topic and sentiment during generation. Alternately, [Chen et al. \(2019b\)](#) proposed disentangling structure and semantics in a multitask framework by introducing losses for paraphrase reconstruction, discriminative paraphrase, and word positions. Along similar lines, in addition to multitasking between content and style, [John et al. \(2019\)](#) introduced adversarial loss to preserve the content while varying style. This discriminative training was, in fact, previously explored by [Hu et al. \(2017\)](#). Hence disentanglement and interpretability is not only an active area of research but also a prospective direction in factorizing the different properties and thereby addressing each according to their nuances.

2.3.1 Anchored Text Generation

This section provides a peek into the modeling approaches in some of the text generation tasks through the lens of the anchoring framework. As we will discuss in this section, the anchors for some of them are mid-way between N-Local and N-Global. Text generation systems are pervasive in everyday AI applications all around us. There are three paradigms of text generation. They are *text-to-text*, *data-to-text* and *none-to-text* paradigms. Moreover, in order to make these universally useful text generation systems more natural, they are usually conditioned with various factors derived from the properties of human communication. In the framework discussed in the previous chapter, generation is anchored in the narrative properties. These high-level properties can be realized with different variants and forms such as emotional, personality-based, or topic-aware generation. This thesis focuses on modeling approaches of utilizing these anchor or conditioning units in deep learning and traditional approaches.

There is a neat synergy between the tasks of language modeling and text generation. Language modeling predicts the next word given the history of the sequence until the current time step. Similarly, text generation models predict the next given the generated sequence so far in addition to some conditioning input.

$$\text{Language Modeling} : p(Y) = \prod_{t=1}^T p(y_t | y_{<t}) \quad (2.17)$$

$$\text{Generation} : p(Y|X) = \prod_{t=1}^T p(y_t | X, y_{<t}) \quad (2.18)$$

Based on the different paradigms, X can take the form of images, text or other data.

Summarization: An important distinction that occurred while shifting from extractive to abstractive summarization is the step of extracting important or relevant information. This is usually taken care of by the encoder. As the size of the document or the number of the documents increases, as in the case of multi-document summarization, it hinders the encoding of important information. Broadly, I present N-Local and N-Global anchoring to address the

problems of content selection, copying, hybridized word, and character modeling and distraction techniques.

- *N-Local Anchoring*: One way to address the aforementioned issue is using a separate content selection network. A very intuitive and elegant solution is proposed by [Gehrmann et al. \(2018\)](#) which is a two-step process. The first step is a data-efficient content selector that determines binary tags based on the presence of words in the final summary. The second step uses this information to constrain the model to generate from likely phrases using a bottom-up attention mechanism. The anchor units are derived from word or phrase level masking and this anchoring is provided locally by providing the words that are predicted to be present in the final summary. Similarly, [Chen and Bansal \(2018\)](#) have introduced a convolutional encoder that computes representation for each sentence. The anchor units are derived by first selecting the entire salient sentences using a pointer network and then using an abstractive layer to translate them through more fine-grained guidance into summaries. Similarly, [Nallapati et al. \(2016\)](#) proposed two approaches that classify whether a sentence is present or not.
- *N-Global Anchoring*: A crucial issue to tackle is that sometimes in summaries we would like to replicate the information from the source document. [See et al. \(2017\)](#) proposes a neat way of copy mechanism based on context vectors from attention. While decoding each word, there is a probability p_{gen} with which the generator word is picked and $1 - p_{gen}$ with which we copy over the word in the form of sum of attention distributions over the source document. This gives the final probability distribution for that time step. The copy probabilities are anchoring into the content by the means of soft combination of word representations from the attention context vectors. This is a means of anchoring that is in midways between N-Local and N-Global anchoring since only a few words are anchored by copying from the source document. While attending to the words that we want to copy is essential, it is also important to look at the rest of the document to get a wider range of content. In this regard, [Chen et al. \(2016\)](#) proposed distraction based techniques. The idea is to distract the model and enable it to navigate the rest of the document in order to assimilate the entire meaning. The entire history context vector and the current context vector determine this distraction at the summary level.

In contrast, [Chang et al. \(2018a\)](#) also address summarization using a hybrid word-character approach that preserves the advantages of both word and character-based representations. This is a plain seq2seq model without any explicit anchoring.

Image Captioning: This task falls in the category of data-to-text generation with data being the images. The main challenges in this task include visual encoding objects, semantic concepts, mapping from language to image regions. The prior work to address these issues can be categorized into both the anchoring mechanism as follows.

- *N-Local Anchoring*: [Xu et al. \(2015\)](#) used stochastic hard and deterministic soft attention on images to perform image captioning. Basically, the model learns alignment between

the visual features and words, and these attention scores are used to decode a caption. This alignment is learnt in an end-to-end fashion without any hierarchical or dual-stage modeling. The guidance from each of the image regions provides a fine-grained guidance to the words in the caption. However, sometimes there are no grounded positions or representations in an image that each word in the caption can attend to. For instance words like “a”, “the”, and also other words that are strongly guided by the language model. Hence, in addition to spatial attention, [Lu et al. \(2017\)](#) use a gating mechanism that was introduced in this work as a visual sentinel to determine when to look at the image. This provides a tradeoff between the support from the image and the confidence of the language model. And an adaptive attention is proposed by attending to the spatially attended image features or the visual sentinel vector to derive the context vector for the current time step in decoding. Essentially the sentinel enables a bridge between N-Local and N-Global anchoring into the spatial regions of the input image. The anchor units here are image regions and the words decoded so far. Similar to the bottom-up summarization seen earlier, [Dai et al. \(2018a\)](#) propose a dual-stage model that first extracts an explicit semantic representation in the form of verbs and nouns from the given image. A set of noun phrases are extracted from captions and binary classification is performed for the presence of each of the noun phrases in an image. The second stage generates the caption in a bottom-up recursive fashion. In this stage, a list of ordered pairs of NPs are maintained. A connecting module attempts to combine these phrases. This provides dense and fine guidance in generating each word. The anchor units are noun phrases which are termed as semantic concepts in this work.

- *N-Global Anchoring:*

[Venugopalan et al. \(2017\)](#) proposed a model to identify parts of the image like objects which can be used to decode a caption, particularly helpful when there is not a lot of annotated data for captions. The visual features are trained for object detection with image-specific cross-entropy loss. Another language model is trained on huge text corpus such as Wikipedia optimized with maximum likelihood. These image-specific and text-specific representations of the previously generated words are added to provide an object-conditioned caption generator. However, the problem of using a pretrained object detector and language model is catastrophic forgetting. So they perform joint training and share parameters between these models along with optimizing for a combined loss. Detecting the object serves as a theme for generating an entire caption. Next, unsupervised image captioning is also done through language pivoting ([Gu et al., 2018b](#)) where two models, the first which is a captioning model and a machine translation model operate in conjunction to generate a sentence in another language where parallel data is unavailable. The model is anchored in the machine translation space.

[Johnson et al. \(2016\)](#) propose dense captioning, where the idea is to generate several sentences corresponding to each of the detected regions in the input image. They introduce a localization layer that does region proposals and thereby region features which are processed with a fully connected recognition network followed by RNN language model to generate captions. With respect to each of the regions, the caption has N-Global anchoring. However, since the model also identifies the regions, with respect to the entire image, the model is anchored locally.

Storytelling: A story is a set of connected events which can be real or fictional that are manifested in various forms (such as pictures, text, signs) and adapting to various cultures. With the increasing success of seq2seq models in capturing long dependencies, the ever-standing goal of generating stories based on human constraints and surroundings has emerged as one of the very interesting tasks in NLP. The conditions can be static or dynamic based on the theme, characters, storyline, the nature of the expected ending etc.,

- *N-Local Anchoring:* As discussed, one of the forms is generating text from images. This is also known as visual storytelling (Huang et al., 2016). The dataset supports the anchoring of each story sentence in the corresponding image caption in addition to the image. Martin et al. (2018) propose a midway abstracted representation of stories in terms of events. An event2event network first identifies the next event given the events so far. This sequence of events is then converted into sentences. Xu et al. (2018a) also introduce a dual-stage skeleton-based story generation where skeleton extraction is performed using sentence compression techniques to identify the most important words. Extensions with respect to event structures are also annotated with a framework to tag causal and temporal relations in stories by Mostafazadeh et al. (2016) that can provide N-Local anchoring for story generation. Also, Tambwekar et al. (2018) apply reward shaping to provide dense rewards for certain related verbs in order to achieve a pre-specified end goal of a story. This also introduces controllability in the sequence of anchors. Yao et al. (2019) also provide static and dynamic plans based on the frequency of words.
- *N-Global Anchoring:* An N-Global anchoring can be provided in terms of the theme of the story. Fan et al. (2018) uses a hierarchical approach to first decode a premise or a prompt for the story. Following this, a modified cold fusion technique is applied to a combination of a pre-trained language model and a trained seq2seq model. Similar to Tambwekar et al. (2018), when a pre-specified ending is given in sparse reward settings, this categorizes as N-Global anchoring.

Dialog Response Generation: Dialog generation is the task of modeling a discourse or conversation between a bot and humans to emulate the natural exchange of ideas or thoughts. This can be goal-oriented or for chit-chat. Often, the conditions from which the anchors are derived are dynamic based on the progress of the dialog.

- *N-Local Anchoring:* Xing et al. (2017) introduce a topic aware modeling technique for dialog response generation. A biased probability distribution of words is computed from a joint attention over the topic words and the vocabulary space from the input messages. By adding the topic words probability, the generation distribution is biased to the topic words. Similarly, Dziri et al. (2019) proposed a topical hierarchical recurrent encoder-decoder model to derive context and dynamic topic attention to decode each word. This topic-aware attention is provided for each time step of the decoding. Hence, these models locally anchor topics in response generation. Zhou et al. (2018c) present an emotional chatting machine that decodes each word based on the probability distribution between the softmaxes over the vocabulary and emotion. The emotion is modeled with an external memory comprising of the emotion-specific words. Similarly, Ghazvininejad et al. (2018)

incorporate world facts and contextual facts before the decoder to build a knowledgeable conversation model.

- *N-Global Anchoring*: Luan et al. (2017) address generation of dialog response in a target personality by proposing a multi-task learning architecture. The primary task is modeled using a seq2seq network with message and response as input and output. Another autoencoder model is trained with the target speaker data. The decoder parameters between both the networks are shared and the model. In this way the speaker role is adapted to response generation by N-Global anchoring using shared decoder parameters. Also, Mou et al. (2016) use point-wise mutual information to derive a keyword, which is usually a noun, and generate the entire response using both forward and backward conditioning on this keyword.

2.4 Multiview Interactions with Anchors

The modeling challenge now boils down to how these anchors (derived from textual modality) interact with the other modalities or views. The secondary view in this case is visual modality or another language. Figure 2.5 presents 3 categories of methods that anchors and other modalities (both textual and visual) interact to generate textual output. For representational purposes, let us consider the secondary view to be visual modality. So, with the input images I , and the output text T , anchors A and predicted anchors A' in text, these main categories are: (a) independent modeling, (b) fusion modeling, (c) latent modeling.

Deriving Anchors: Identifying appropriate anchors corresponding to these properties is essential to bring forth their manifestation in the generation. These anchors are often not readily available or are paired with the data. Hence we resort to the following weakly supervised noisy approximates including (i) training on external data and using the resulting model to predict weak labels, (ii) using off-the-shelf tools to weakly annotate the required anchors (iii) deriving anchors from context representations.

Independent Anchoring: The individual views are self-governed and their representations are not subjected to control by the other views. Scoping the definition means that the textual anchors are not available along with visual inputs. In such circumstances, we can utilize the weakly annotated anchors only in the output. The first setup is without the use of any forms of anchors where only the images are used to generate the text ($I \rightarrow T$). To improve the optimization for the representation of the anchors, the second setup multitasks with the prediction of the anchors as an auxiliary task for generating the text. And finally, this is extended to the third setup with a hierarchical multitasking model where the predicted anchors are leveraged to generate text along with the images.

Fusion Anchoring: In this category, both the views i.e., anchors and images are modeled together. The multimodal representation fusion transforms the independent representations from these different views to form a super representation/vector with combined concatenated

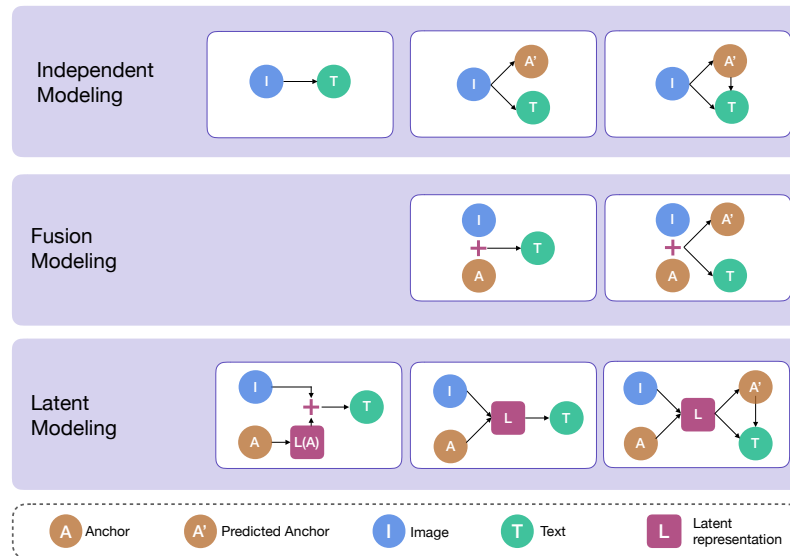


FIGURE 2.5: Interactions among multiple views (visual modality with textual anchors)

information. The early fusion occurs at the feature representation stage where these modalities are directly combined before the actual modeling. The late fusion methods utilize these representations independently and combine them in the decision making stage. In addition, the fusion methods can also be combined with the multitasking or hierarchical multitasking that we discussed above.

Latent Anchoring: Instead of directly using the representations of the anchors, they can be modified based on selective cross attention based on the context or the images. This latent vector is then used in any of the techniques mentioned above including fusion modeling, multitasking etc.,

The rest of the thesis presents the utility of these methods to anchor content, structure and surface form properties of narratives. We will be revisiting the comparison of these methods for various tasks.

2.5 Conclusions

The past decade witnessed text generation dribbling from niche scenarios into several mainstream NLP applications. This urges the need for a snapshot to retrospect the progress of varied text generation tasks in unison. In this chapter, we highlighted some of the existing challenges in each of the three main task-agnostic components of generation: training paradigms including generative pre-training, decoding strategies, and evaluation. In addition, this section also includes a dense account of key challenges along with some techniques to address them. We believe this section manifests as a compact resource for major outstanding challenges in this vast maze of neural text generation that can assist researchers foraging to situate their work in a task-agnostic manner to accelerate the progress in the future. We believe understanding

this space helps us foresee upcoming challenges in context by broad audiences, researchers and practitioners, in academia and industry. Moving forward, we envision that there are some crucial directions to focus on for impactful innovation in text generation. These include (i) *generation in real-time* (ii) *non-autoregressive decoding* (iii) *grounding with situated contexts in real and virtual environments and games* (iv) *consistency with personality and opinions especially for virtual agents* (v) *conditioning on multiple modalities together with text and data* (vi) *investigating better metrics to evaluate generation to correlate better with human judgments* (vii) *creative text generation, such as jokes, sarcasm, metaphors etc.*, (viii) *utilizing large scale pre-trained language models as knowledge sources*, (ix) *reducing societal biases and generating text ethically*, (x) *setting up benchmarks to evaluate generation agnostic and specific to individual tasks*. We believe this is the right time to extend advancements in any particular task to other tightly coupled tasks in order to revamp improvements in text generation as a holistic task in the future.

An upsurge in the interest in virtual beings invites research questions and opportunities towards building a generative environment. Such virtual conscious characters capable of understanding, narrating, and responding in real-time have immensely useful applications such as digital education, assisting the visually impaired, collaborative authoring etc., Automated generative narratives are a direct consequence of an explosive complexity of state space that is laborious for humans to express. It merely does not encompass the quantity but also opens up doors to diversified quality with rich personalized requirements and interactions. Existing technologies are capable of dealing with a subset of such diversities and demand continual work towards developing distinctive interactions that make a difference in the narrative. This broad goal also appeals to different modes of interactions with these virtual characters in the future. This thesis is a language technologies based step towards achieving this multiview goal. I believe this thesis is scratching the surface of an exceptionally potential field of study to assist humans.

Part II

Anchoring Narrative Properties

3

Content in Narratives: N-Local and N-Global Anchoring

The number one factor in engagement is relevance, because relevance drives out resistance.

Clive Shepherd

Content units are the basic building blocks of a narrative, and the way these units are arranged determines the structural property of the narrative. Generating text with appropriately relevant content is the first and foremost requirement to communicate our intent. Hence fundamentally identifying the relevant content that acts as underlying bolsters is the core part of presenting the appropriate and relevant information. This information can be either in the form of a skeleton indexing into every unit of the narrative or probing and providing a general theme for the narrative. The former is N-Local anchoring, and the latter is known as N-Global anchoring.

Formally, N-Local anchor units are indexed into every unit of the narrative individually and explicitly. Hence there is a one to one mapping for each of $\alpha_i^{(j)}$ to each of the generated

This chapter is based on the following papers:

- “*Reading Between the Lines: Exploring Infilling in Visual Narratives*” (Chandu et al., 2020a)
- “*Denoising Large-Scale Image Captioning from Alt-text Data using Content Selection Models*”(Chandu et al., 2020b)
- “*Induction and Reference of Entities in a Visual Story*” (Dong et al., 2019c)
- “*Tackling Biomedical Text Summarization: OAQA at BioASQ 5B*” (Chandu et al., 2017a)

sentence $s_i^{(j)}$. These anchor units are fine-grained to each and every unit of generation. To incorporate this, there is a bijective function between anchor units and generated text. Given I_i and A_i , the task is to generate N_i .

The anchors A_i can take several forms facilitating the content in the narratives. In the case of N-Local anchoring, we will be looking into inducing entities in the task of visual storytelling in a supervised learning setting. The anchoring here is done via entity skeletons. Anchors are represented in three forms: *surface forms*, *nominalized forms*, *abstract forms*. This is addressed using hierarchical attention-based models to anchor the entity skeletons in different levels.

Similarly, for N-Global anchoring, N_i which contains $s_i^{(k)}$ is assembled from an anchor A_i which is not further split to k units to provide finer level guidance. In this case of N-Global anchoring, a theme or topic is provided for the whole narrative and not directly for each sentence.

N-Global anchoring is examined in the case of topic-based summarization, where the topic is derived from a query. This topic dictates the N-Global anchor of the summary, which determines the extraction of relevant content sentences. In this case, there are three forms of representing the anchors that comprise of *surface forms*, *expansion forms*, *embedding forms*.

In this chapter, we will have a closer look at each of these anchors based on granularity levels. As mentioned above, the focus is not deriving these anchor points but utilizing them in aggregating a narrative. The rest of the chapter is organized as follows. In §3.1, we will discuss the related work in the domains of narrative generation, particularly visual storytelling and biomedical summarization, in the perspective of N-Local and N-Global anchoring. §3.2 describes how anchors can be used to denoise large-scale image captioning, demonstrating the cross-lingual transferability and controllability that they offer. This is followed by §3.3 which delves into details of anchoring N-locally or in a fine-grained fashion for generating stories from visual input and image captions. Subsequently, in §3.5, a query-based summarization system is discussed to derive the content based on the elements in the query. Finally, §3.6 concludes this chapter and describes prospective research directions that can be taken up in this domain.

3.1 Related Work

Multimodal Language: Language generation from visual modality has seen a steep rise in interest with the introduction of several large scale tasks such as image captioning (Hossain et al., 2019), visual question answering (Antol et al., 2015) and visual dialog (Das et al., 2017; Mostafazadeh et al., 2017; De Vries et al., 2017). While the task of generating a sentence from a single image, i.e., image captioning has been well studied in the literature, generating a long-form sequence of sentences from a sequence of images has been catching attention only in the recent past. Hence, the natural next step is towards long-form sequential generation in the form of stories, procedures, etc., visual narrative telling.

Visual Storytelling: Research at the intersection of language and vision is accelerating with tasks like image captioning (Hossain et al., 2019), visual question answering (Wu et al., 2017), visual dialog (Das et al., 2017; Mostafazadeh et al., 2017; De Vries et al., 2017; de Vries et al., 2018). Huang et al. (2016) ventured into sequential step-wise generation of stories by introducing visual storytelling (ViST). Recent methods have tackled ViST using adversarial learning, reinforcement learning (Wang et al., 2018; Huang et al., 2019c; Hu et al., 2020b), modality-fusion (Smilevski et al., 2018), traditional seq2seq models (Kim et al., 2018; Jung et al., 2020; Hsu et al., 2018) and explicit structures (Bosselut et al., 2016; Bisk et al., 2019). Chandu et al. (2019a) also proposed a dataset of 16k recipes in a similar form. While these are all cooking recipes, the ViPT dataset comprises a mixture of ten different domains. Also, our dataset is about 2.8 times larger than the storyboarding dataset, with almost double the number of procedures in the domain of cooking recipes itself. Though the stories in ViST demonstrate a sense of continuity, the overarching sequential context is feeble. Procedures such as cooking recipes (Salvador et al., 2019; Wang et al., 2019b) on the other hand, demonstrate this characteristic inviolably. This ensures a coherent underlying context and structure in the narrative. Hence, we present a large-scale ViPT dataset to encourage research in this direction.

Conditioning on raw input: Most of the foundational techniques used to address this task rely heavily on seq2seq models. In such approaches, anchoring is a proxy for conditioning on the raw input images. Kim et al. (2018) proposed a seq2seq framework that takes in the raw images from which ResNet features are extracted to model the story text corresponding to each individual image. Smilevski et al. (2018) proposed late fusion techniques to address this task. We derive motivation from these techniques to introduce entities and references as skeletons.

Anchoring Content in Generation: Grosz et al. (1995) was one of the initial works delving into how entities and their referring expressions are used in a discourse context. Several research efforts for narrative generation tasks have spawned from introducing a schema or a skeleton. Martin et al. (2018); Clark et al. (2018b) explored the usage of event representations and predicting successive event forms to generate the entire story. Fan et al. (2018) proposed hierarchical frameworks for story generation conditioned on a premise or a topic. This work was also extended by decomposing different parts of the model by generating a surface realization form of the predicate-argument structure by abstracting over entities and actions (Fan et al., 2019a). Xu et al. (2018a) used reinforcement learning first to generate a skeleton (the most critical phrases) and then expand it to a complete sentence. Yao et al. (2018a) proposed a hierarchical generation framework in which, given a topic, the model first plans a storyline and then generates a story based on the storyline. Recently, Zhai et al. (2019) proposed a model to generate globally coherent stories from a fairly small corpus by using a symbolic text planning module to produce text plans and then generating fluent text conditioned on the text plan by a neural surface realization module. Ammanabrolu et al. (2019b) showed that event-based generation often generated grammatically correct but semantically unrelated sentences and present ensemble methods for event based plot generation as a solution.

Content selection from vision: There is a rich body of work in improving content selection for IC (Feng et al., 2019), mainly focused on scene graph based skeletons (Gu et al., 2019b; Kim et al., 2019b; Chen et al., 2020a; Yang et al., 2019c). However, these annotations with objects

and relations are expensive, thereby constraining the scaling up to multiple languages and diverse concepts. Our work delegates this responsibility of identifying content to the language modality by using inexpensive, off-the-shelf tools for weak supervision.

Content selection from language: An orthogonal body of work relies on skeletons derived from language using hierarchical phrase modeling (Tan and Chan, 2016; Dai et al., 2018b), semantic attention (You et al., 2016), attribute LSTM (Yao et al., 2017), skeleton-based attribute filling (Wang et al., 2017), adaptively merging topic and visual information (Liu et al., 2018a), multimodal flow (Li et al., 2019a) and concept guided attention (Li et al., 2019b). Note that all these prior works utilize human-curated gold datasets such as COCO (Lin et al., 2014) and Flickr30k (Plummer et al., 2015a) with clean coupling between captions and images. However, scaling them to large and diverse concepts is expensive. We utilize *uncurated* silver standard datasets with the advantages of richness and diversity at the cost of noisy text. Hence we show the effectiveness of a dual staged approach that denoises the captions by skeleton prediction.

Cross-lingual and controllable captions: Past work on cross-lingual captioning focused on translation (Barrault et al., 2018), fluency guidance (Lan et al., 2017), using large datasets (Yoshikawa et al., 2017) and more recently by pivoting on source language captions (Thapliyal and Soricut, 2020; Gu et al., 2018c). We go a step further and pivot on the predicted English skeleton to improve multilingual captions due to a dearth of similar off-the-shelf tools in other languages. We qualitatively explore controlling length via skeletons which was explored before via adding length to decoder (Luo and Shakhnarovich, 2020; Cornia et al., 2019). Other controllable aspects include stylistic captions (Guo et al., 2019b; Mathews et al., 2018) language (Tsutsui and Crandall, 2017) which are potential extensions to our unpaired captioning work.

Interpretable Natural language skeletons: Despite remarkable advancements of large scale end-to-end models, recent work identifies spurious correlations in the datasets that potentially leads to high performances (Geva et al., 2019; Tsuchiya, 2018). To mitigate this, researchers began dissecting intermediate components of the models with the goal of interpretability to humans (Wiegrefe and Pinter, 2019; Thorne et al., 2019; Lipton, 2018) as opposed to implicit explanation (Xu et al., 2015). Our work can also be viewed as an instance of explaining captions through skeleton predictions similar to recent works on rationalizing answer predictions for question answering (Latcinnik and Berant, 2020). We view this interpretable intermediate layer as a peek into the model predictions helping us study more subtle but crucial dataset attributes, such as gender bias, and provide human-in-the-loop interventions to improve the final caption.

Infilling and Masking: The idea is motivated by cloze tasks (Taylor, 1953) that address readability and understanding. However, recent advances in learning a masked language model (Devlin et al., 2019b) paved the way for a new trend in exploring masked contexts (Song et al., 2019; Lewis et al., 2019). Generation of meaning patches with missing portions of text is experimented by Zhu et al. (2019); Donahue et al. (2020); Fedus et al. (2018b) to generate meaningful patches. Similarly, Ippolito et al. (2019) proposed a hierarchical model to generate middle span using a bag of predicted words from left and right contexts. In a similar spirit, this paper studies the effects of infilling techniques for visual narrative generation. An alternate stream of work to improve the context in stories include providing supporting information

such as entities (Clark et al., 2018b; Xu et al., 2018a), latent templates (Wiseman et al., 2018), knowledge graphs (Yang et al., 2019b), etc., explicitly. In contrast to this, infilling provides an opportune platform to learn contextual information implicitly. Our work is positioned at the intersection of infilling and multimodal language generation.

N-Local Anchoring in Visual Storytelling: Park and Kim (2015) proposed a coherent recurrent convolutional network that utilizes an entity-based local coherence model that resolves local transitions. They prepare an entity grid with a cross product between the sentences and the available discourse entities with annotations on the grammatical role. In this way, there is N-Local anchoring of entities in generating descriptions for a sequence of images with fine-grained guidance. There has been other work by Liu et al. (2017b) who explored the task of generating a sequence of sentences for an image stream by anchoring in the joint semantic space. The joint here represents the multimodal joint space of vision and language. In addition to conditioning the input image itself from the sequence of images, each step is anchored in the semantic space that draws correspondences between the image and sentence embeddings by minimizing contrastive loss between them. This model leverages the semantic coherence in a photo sequence with a bidirectional attention-based recurrent model to generate stories from images.

Our work in N-Local anchoring in visual storytelling falls along the lines of generating a story from visual input based on schema.

Biomedical Summarization: The problem of extracting *exact answers* for factoid questions from this data is being studied extensively, resulting in the development of several techniques, including inferencing (Moldovan et al., 2002), noisy-channel transformation (Echihabi and Marcu, 2003) and exploitation of resources like WordNet (Lin and Hovy, 2003). However, recent times have also seen an interest in developing *ideal answer* generation systems that can produce relevant, precise, non-repetitive, and readable summaries for biomedical questions (Tsatsaronis et al., 2015).

N-Global Anchoring in Content Selection: A query-based summarization system called “BioSQUASH” Shi et al. (2007) uses domain-specific ontologies like the Unified Medical Language System (UMLS) (Schuyler et al., 1993) to create a conceptual model for sentence ranking. Experiments with biomedical ontology-based concept expansion and weighting techniques were conducted, where the strength of the semantic relationships between concepts was used as a similarity metric for sentence ranking (Chen and Verma, 2006). Similar methods (Yenala et al., 2015; Weissenborn et al., 2013) are used for this task where the difference lies in query similarity ranking methods. Our work focuses on anchoring the content through novel similarity computation to gather the content anchored in the question. Fan et al. (2018) adopt a hierarchical approach to generate a premise and then stories to improve coherence and fluency where the narration is driven from N-Global anchoring in the theme or the premise.

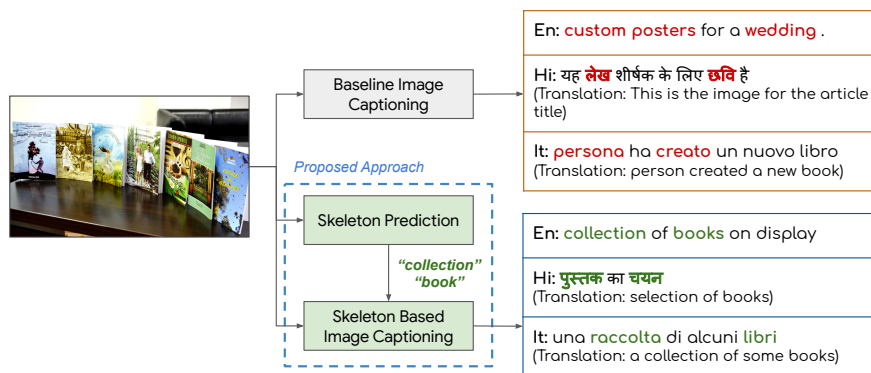


FIGURE 3.1: Overview of our approach: (1) skeleton prediction & (2) skeleton based IC; compared to conventional IC. Output captions shown in English (En), Hindi (Hi) and Italian (It).

3.2 Content selection for Denoising Image Captioning

Training large-scale image captioning (IC) models demands access to a rich and diverse set of training examples that are expensive to curate in terms of time and manpower. Instead, alt-text-based captions gathered from the web are a far cheaper alternative to scale with the downside of being noisy. Recent modeling approaches to IC often fall short in terms of performance in leveraging these noisy datasets in favor of clean annotations. We address this problem by breaking down the task into two simpler, more controllable tasks – skeleton prediction and skeleton-based caption generation. Specifically, we show that *sub-selecting content words as skeletons* helps in generating improved and denoised captions when leveraging rich yet noisy alt-text-based *uncurated* datasets. We also show that the predicted English skeletons can further cross-lingually be leveraged to generate non-English captions and present experimental results covering caption generation in French, Italian, German, Spanish, and Hindi. We also show that skeleton-based prediction allows for better control of certain caption properties, such as length, content, and gender expression, providing a handle to perform human-in-the-loop interpretable semi-automatic corrections.

In the last demi-decade, most of the NLP fields ventured into reaping the benefits of utilizing large-scale raw data (*uncurated*) from web crawls. This trend resonated with new uncurated image-captioning datasets like Conceptual Captions (Sharma et al., 2018). While these uncurated alt-texts are superior in terms of size and diversity in the dataset, they are inferior to the well-curated datasets (Lin et al., 2014; Wang et al., 2019c) in terms of noisiness in the captions. The content in the alt-text for the image is often distorted in favor of the intent or the context in which the image is presented. For example, the ground truth alt-text caption for a house is ‘house for sale’ instead of ‘front view of a house’. This noise hinders exploiting these huge datasets to the fullest.

We present a simple two-staged approach by separating the content selection from caption generation as illustrated in Figure 6.1. In contrast to most IC approaches (Hossain et al., 2018; Sharma et al., 2020), which hallucinate incorrect content from noisy training data (i.e ‘custom posters’ and ‘wedding’), our approach first focuses on *denoising* the content words (i.e ‘collection’ and ‘book’) that are further used to generate a relevant caption. We refer to this sequence of concept words that are key pieces of information consistent with the image as a *skeleton*.

Sub-selecting skeleton words that curb noisiness are automatically extracted from the alt-text captions. We focus on language-based skeletons that are derived from captions (Kuznetsova et al., 2014; Fang et al., 2015; Dai et al., 2018b), rather than expensive visual-based skeletons derived from image, e.g., scene graphs, (Wang et al., 2019a; Yang et al., 2019c), which are hard to scale. More concretely, we introduce an intermediate task of distantly supervised skeleton prediction in the end to end IC pipeline: The end-to-end task of IC is $(f_\theta : \mathbb{I} \rightarrow \mathbb{C})$ is broken down into a dual-staged pipeline: skeleton prediction $(f_\theta : \mathbb{I} \rightarrow \mathbb{S})$ and skeleton based captioning $(f_\phi : \mathbb{I}, \mathbb{S} \rightarrow \mathbb{C})$, where \mathbb{I} is the image, \mathbb{S} is the skeleton, and \mathbb{C} is the caption (Kulkarni et al., 2013; Li et al., 2011; Elliott and Keller, 2013; Fang et al., 2015). We present a comparison between encoding, decoding, and autoencoding these skeletons. As such, our skeleton prediction solution addresses the *semantic gap* problem (Li and Chen, 2018; Yao et al., 2018b).

We illustrate the effectiveness of this approach on noisy uncurated datasets in the following ways. (1) We demonstrate that sub-selecting content words with an intermediate *skeleton prediction task denoises content* thereby leading to better human evaluation results on captioning. We also conduct extensive analysis on multimodal discourse relations to understand the reasons for this improvement (Alikhani et al., 2020) being the generation of more visible captions. (2) Scaling the large uncurated datasets to other languages is still a bottleneck. We show the *transferability of learning English skeletons* to improve caption generation in other languages – English, French, Italian, German, Spanish, and Hindi. (3) The predicted skeletons qualitatively demonstrate other potential benefits, such as *controllability* of content, length, and gender via a natural language-based *interpretable* interface, which enables one to additionally interact with the generation process.

3.2.1 Datasets Description

Conceptual Captions (CC): CC (Sharma et al., 2018) is a large-scale dataset of 3.3M image-caption pairs covering a large variety of processed alt-texts from the web. The focus of this work is on denoising noisy captioning datasets (web-scale, not human-verified). Hence our experiments are focused on CC, which is a step closer to having large and diverse alt-texts from the web at the cost of being noisy. In contrast, other popular datasets like COCO (size 120K) (Lin et al., 2014) and Multi30k (Elliott et al., 2016) are hand-annotated by humans and contain high quality images/captions. As a resource, CC is useful both for measuring progress on large-scale automatic captioning (Sharma et al., 2018; Changpinyo et al., 2019; Alikhani et al., 2020; Thapliyal and Soricut, 2020), as well as pre-training data for a variety of vision-and-language tasks (Lu et al., 2019a; Chen et al., 2020d; Tan and Bansal, 2019; Su et al., 2020; Li et al., 2020a).

Pre-processing: CC might contain a long tail of spelling errors and other typos due to the automatic curation of the data. Therefore, we perform frequency-based thresholding of the skeleton words to abate this noise. We experimented with several values for this hyperparameter and selected a minimum occurrence count of 50, providing the desired balance between noise and vocabulary size.

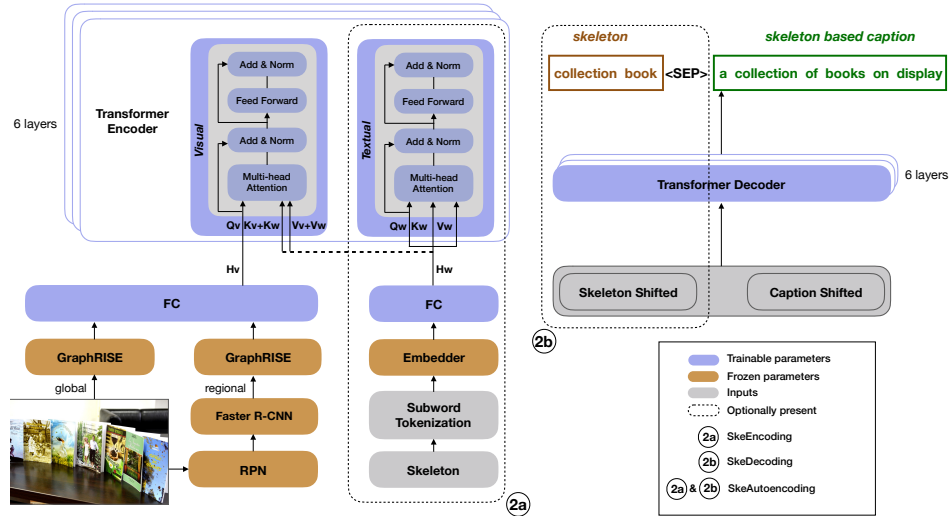


FIGURE 3.2: Model architecture of our skeleton based captioning along with *text as side attention* mechanism between visual (v) and textual (w) modalities. The skeleton is present optionally in the encoder, decoder or both based on our three approaches.

Multilingual CC: To demonstrate the cross-lingual transferability of our skeletons, we use automatic caption translations¹ for CC, similar to the approach in (Thapliyal and Soricut, 2020). Note that the skeletons are learned from and predicted in English (not in the final target language), making the English skeleton act as an *interlingua*. Since multilingual captions are all pivoted on English skeletons, this nullifies the requirement to 1) collect large-scale image-caption pairs in various languages, and 2) have access to linguistic tools to analyze captions in each language. We perform experiments on five languages – French, Italian, German, Spanish, and Hindi – which vary in word orders and token overlap with the English skeletons.

Conceptual Captions T2 test set: For human evaluations across *all languages*, we use T2 test set used in the Conceptual Captions Challenge². It comprises of 1K out of domain images from the Open Images Dataset (Kuznetsova et al., 2020).

3.2.2 Models Description

IC requires paired examples of images and captions (\mathbb{I}, \mathbb{C}), where $c \in \mathbb{C}$ correspond to tokens in a caption (c_1, c_2, \dots, c_m) , that are often expensive to gather. In contrast, our approach uses intermediate skeletons as an effective way to leverage noisy, uncurated alt-text-based captions to train a model to generate more visually informative captions. An overview of both the stages is presented in Fig. 6.1.

¹We use the Google Cloud Translate API.

²<http://www.conceptualcaptions.com/>

3.2.2.1 Distantly Supervised Skeletons

Since gold standard skeleton words are usually unavailable for IC datasets, we use distant supervision to get these labels. We retrieve syntax annotations (specifically parts-of-speech (POS) and word lemmas), using the Google Cloud Natural Language API ³ over the caption texts. We use these annotations to experiment with the following four variants of skeletons.

1. *Nouns & Verbs*: This includes a sequence of lemmas of all the nouns and verbs in a caption.
2. *Salient Nouns & Verbs*: The saliency of nouns and verbs is determined using tf-idf scores, treating each caption as a document. For each caption, the top 2 highest scoring noun and verb tokens (lemma) are selected. This examines if saliency contributes towards the effectiveness of the skeleton.
3. *Nouns*: This includes lemmas of all the nouns. This helps us untangle the roles of nouns vs. verbs in the effectiveness of the skeleton.
4. *Iteratively refined captions*: Under this condition, the output of the baseline Img2Cap model serves as the ‘skeleton’ for the next skeleton-based captioning stage. The rationale behind this skeleton is to compare the utility of sub-selecting skeleton words based on POS in denoising caption content compared to a full caption prediction.

We ignore skeleton tokens with a frequency of less than 50 in our training data to reduce noise. This subselection of content based on POS tags and downscaling of vocabulary helps retain important words as skeletons resulting in a label size of 5k.

3.2.2.2 Modeling

Baseline (Img2Cap): We adopt an encoder-decoder ($f_\theta : \mathbb{I} \rightarrow \mathbb{C}$) IC model based on Transformers (Vaswani et al., 2017a) following recent state-of-the-art approaches (Sharma et al., 2018; Yu et al., 2019; Changpinyo et al., 2019; Huang et al., 2019a; Cornia et al., 2020). Our model uses the IC framework introduced in Changpinyo et al. (2019). Inspired by the bottom-up and top-down approach (Anderson et al., 2018a), the input image \mathbb{I} is represented as a bag of features containing one global and 16 regional, fine-grained feature vectors. The regional features correspond to the top 16 box proposals from a Faster-RCNN (Ren et al., 2015) object detector trained on Visual Genome (Krishna et al., 2017), with a ResNet101 (He et al., 2016a) that is trained on JFT (Hinton et al., 2015) and fine-tuned on ImageNet (Russakovsky et al., 2015a). We featurize both global and regional boxes using Graph-RISE (Juan et al., 2019, 2020). We make the following changes to the state of the art model (Changpinyo et al., 2019), leading to a 9-point improvement on the dev CIDEr on CC (1.00 vs. 0.91) (**improved baseline**): 1) encode the corners and the area of the bounding boxes to fuse positional information with visual features, (Lu et al., 2019b), and 2) encode each feature vector with a Linear-ReLU-LayerNorm-Linear instead of Linear embedding layer, where LayerNorm is layer normalization (Ba et al., 2016).

³<https://cloud.google.com/natural-language>

	Stage 1		Stage 2		Conditioning
	Input	Output	Input	Output	
SkeEnc	\mathbb{I}	\mathbb{S}'	$\mathbb{I} + \mathbb{S}'$	\mathbb{C}'	$\tau \sim \prod_t Pr(\tau^{<t}, g(z_{\mathbb{I}}, \hat{\mathbb{S}}))$
SkeAE	\mathbb{I}	\mathbb{S}'	$\mathbb{I} + \mathbb{S}'$	$\mathbb{S}' + \mathbb{C}'$	$\tau \sim \prod_t Pr(\tau_k^t [\hat{\mathbb{S}};^{<t}], g(z_{\mathbb{I}}, \hat{\mathbb{S}}))$
SkeDec	(no Stage 1)		\mathbb{I}	$\mathbb{S}' + \mathbb{C}'$	$\tau \sim \prod_t Pr(\tau_k^t [\hat{\mathbb{S}};^{<t}], z_{\mathbb{I}})$

TABLE 3.1: The inputs and outputs of the different models. In iterative refinement, \mathbb{S}' is replaced by \mathbb{C}' .

Dual Staged Modeling: In this approach, we introduce an intermediate natural-language interpretable skeleton \mathbb{S} between \mathbb{I} and \mathbb{C} . This \mathbb{S} is composed of a sequence of lemmas, using a subset of content words (s_1, s_2, \dots, s_n) from c , where $n < m$. This reduces the output complexity of $f_\theta : \mathbb{I} \rightarrow \mathbb{C}$ by simplifying and denoising the noisy \mathbb{C} to \mathbb{S} . Hence, the task of IC is decomposed into the first stage of predicting skeleton concepts and the second stage of caption generation using the intermediate skeleton.

Stage 1: Skeleton Prediction (Img2Ske): The first stage ($f_\theta : \mathbb{I} \rightarrow \mathbb{S}$) is to predict one of the 4 variants of the skeleton words (from §3.2.2.1) from the images. We experiment with both classification and generation paradigms that respectively do not possess and possess linear conditioning of the predicted skeleton word on the following words. We observe that the generation-based skeleton prediction results in skeleton words that co-occur in a sentence. In contrast, the classification approach predicts skeleton words relevant to an image like *person*, *man*, *singer* that do not necessarily co-occur in a caption.

To improve co-occurrence of the predicted skeleton words, we generate the skeleton words $\hat{\mathbb{S}}$ autoregressively where each word is conditioned on the previously predicted skeleton word. This conditional dependence models word co-occurrence more tightly as $p(\hat{\tau}_j | I, \hat{\tau}_{<j})$, making the skeleton a sequence of words. The model is optimized with cross-entropy loss, trained using teacher forcing. An attractive property is that the same architecture can be used to decode both the skeleton \mathbb{S} and the caption \mathbb{C} . Moreover, the output tokens predicted in this stage are interpretable, and they are used to condition the second stage of our model.

Stage 2: Skeleton-based Caption Generation: The second stage of training uses both images and skeletons to generate captions $f_\phi : \mathbb{I}, \mathbb{S} \rightarrow \mathbb{C}$. We experiment with three variants of conditioning predicted skeletons via encoding, decoding, and autoencoding as shown in the overall model architecture in Fig. 4.2. The inputs, outputs for each stage, and the conditioning of attention for transformer decoder are compared in Table 3.1.

2a. SkeEncoding: The predicted skeleton from the previous stage is used as input to the encoder. The image encoding and skeleton embeddings are fused with a unidirectional attention mechanism, called **text-as-side** (notated as g). In other words, we use the text representation as “side information” – each transformed image feature unit can attend to other image feature units (self-attention) and text (cross-attention), but the text cannot attend to the image. As shown in Fig. 4.2, this model has the dotted box in the Transformer encoder side, with the textual query, key, value (Q_w, K_w, V_w) and the visual counterpart attending to textual or visual key and value $(K_v + K_w, V_v + V_w)$ with a visual query (Q_v) . We focus on the text-as-side

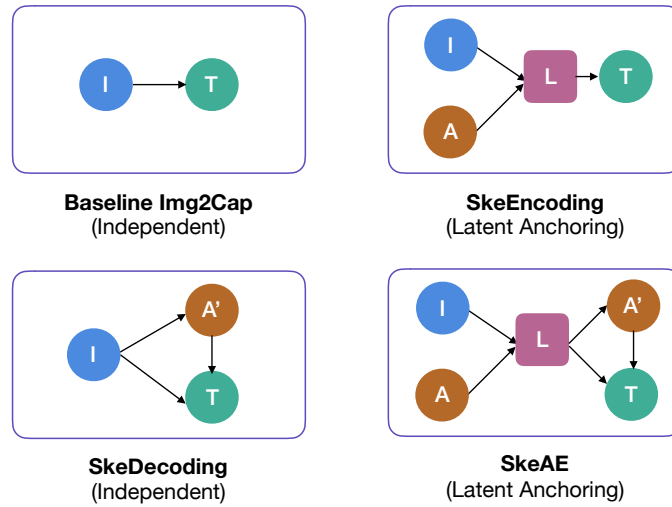


FIGURE 3.3: Comparison of the content-anchoring based on multimodal interactions in denoising

attention mechanism as our preliminary results indicate that it leads to qualitatively better captions than image-text co-attention (Lu et al., 2019a).

2b. SkeDecoding: The skeleton and caption are concatenated and predicted by the same decoder. This is not a two-staged model, as the model is trained to predict both skeleton and caption auto-regressively. The model first predicts the skeleton words conditioned on the previously generated skeleton words. Then every token in the decoded caption attends to the entire predicted skeleton as well as the tokens of the caption decoded until that time step. The dotted box in Transformer decoder of Fig. 4.2 depicts this approach.

2c. SkeAE: To bring both the above models together, we simultaneously encode and decode the predicted skeleton. This brings the benefits of bidirectional attention on the input features (image and predicted skeleton words) and autoregressive attention on the re-predicted skeleton words while generating the caption. In this case, both the dotted boxes on the encoder and decoder sides in Fig. 4.2 are active. The encoding mechanism follows the g function, and the decoder prepends the caption generation task with the predicted skeleton.

The comparison across these models is condensed to their bare forms and presented in Figure 3.3.

3.2.3 Experiments and Results

Hyperparameters: Our transformer model uses six encoder and six decoder layers (unless specified otherwise), with eight heads for multiheaded attention. Captions are subword-tokenized with a vocab size of 8,300. The models are optimized with Adam and an initial learning rate of $3.2e^{-5}$. We use mini-batches of size 128, and train for 1M steps. The token embedding and filter sizes are both 512.

	Iterative Refinement	Classification	Generation
Precision	35.75	23.22	36.66
Recall	24.29	41.31	24.30
F-score	28.92	29.73	29.23

TABLE 3.2: Performance of skeleton prediction stage. Note that for classification and generation, the skeleton type used is ‘nouns & verbs’.

Model	CIDEr		
Baseline (SOTA model)	0.91	(Changpinyo et al., 2019)	
Impr. Img2Cap	1.00		
Impr. Img2Cap (large)	0.99		
Skeleton-based	Skeleton Type		
	Nouns & Verbs	Nouns only	Sal. Nouns & Verbs
SkeEncoding	0.99	0.97	0.94
SkeDecoding	0.99	0.99	0.96
SkeAE	0.99	0.96	0.94

TABLE 3.3: Automatic metrics to compare various skeleton forms. Img2Cap is the baseline (*large* version refers to 12 encoder and decoder layers). Note that these results use generation-based skeleton prediction.

3.2.3.1 Automatic Evaluation

Skeleton Prediction: The goal of this stage is to extract key skeleton words from the image. We compute precision, recall, and F-score as shown in Table 3.2. With the same labels (skeleton: nouns & verbs), both classification and generation approaches have similar F-scores. However, precision is higher for generation, and recall is higher for classification-based predictions. Based on both qualitative observations and human judgments, we note that the generation approach was better, which shows that a higher precision is favorable compared to recall for this stage. The label size (of skeletons) in Table 3.2 is approximately 5K.

Skeleton-based Caption Generation: We report multilingual IC performance of baseline and our dual-stage models using CIDEr in Table 3.3 (English) and Table 3.4 (multilingual). Automatic metrics for captioning are based on surface n-grams, and are not suitable to evaluate when the ground truth captions themselves are noisy. In addition, we find that CIDEr is misleading (Alikhani et al., 2020; Sharma et al., 2018; Seo et al., 2020) and does not correlate with human evaluations (§3.2.3.2).

Multilingual captioning: Note that the skeletons are always in English, trained using annotations over the original English CC dataset. Cross-lingual results on validation data of Multilingual CC are presented in Table 3.4. In addition to the data noisiness, a reason for slightly lower performance for non-English captions is probably noisy translation artifacts. For example, corresponding caption in the Hindi dataset for English caption ‘She is gazing at the *fall colors*’ is *vah girte rangoM ki Or deK rahI hai* (translation: She is looking at the *falling colors*.) Translation errors (such as ‘fall’ colors to ‘falling’ colors) introduce noise in the non-English datasets. Figure 3.4 presents an example of multilingual output captions for the baseline and our SkeAE approach.

Language	Baseline	SkeEncoding	SkeDecoding	SkeAE
French	0.91	0.90	0.89	0.90
Italian	0.90	0.88	0.86	0.87
German	0.74	0.72	0.72	0.73
Spanish	0.92	0.91	0.89	0.91
Hindi	0.85	0.83	0.82	0.82

TABLE 3.4: CIDEr scores for skeleton (form: Nouns & Verbs, prediction approach: generation) conditioned caption generation for multiple languages.

Model Enc Input	CIDEr
PredSke + Img (Paired)	0.99
PredSke (Unpaired)	0.91
GtSke + Img (Paired Headroom)	4.62
GtSke (Unpaired Headroom)	4.48

TABLE 3.5: Ablations on val data for unpaired captioning.


Image	Model	English	French	Italian	German	Spanish	Hindi
	Baseline	spring is in the air	fleurs les plus chères du monde (meaning: most expensive flowers in the world)	un campo di tulipani in primavera (meaning: a field of tulips in spring)	Frühling ist in der Luft (meaning: spring is in the air)	La primavera está en el aire (meaning: spring is in the air)	वसंत हवा में है (meaning: spring is in the air)
	SkeAE pred skeleton: 'tulip field'	pink tulips in a field	tulipes roses dans les jardins (meaning: pink tulips in the garden)	genere biologico in un campo (meaning: biological genus in a field)	ein Feld von rosa Tulpen (meaning: a field of pink tulips)	tulipán en un mar de tulipanes (meaning: tulip in a sea of tulips)	गुलाबी ट्यूलिप का एक क्षेत्र (meaning: a field of pink tulips)

FIGURE 3.4: Captions generated by baseline and our dual staged approach in 6 languages and their corresponding translations.

Unpaired Image Captioning: A natural extension to our approach is for the caption generator to rely purely on predicted skeleton, and not use image features. This is a harder problem but eliminates the need for image-caption pairs altogether because the second stage (skeleton to caption) can be trained on a large text-only corpus. In this direction, within the scope of CC dataset, we investigate 1) with and without using image features in the second stage, 2) using ground truth skeleton (GTSke) to get an estimate of the upper bound on unpaired captioning 3) comparing the upper bound to the predicted skeleton (PredSke). These results are presented in Table 3.5. When image features are ignored, CIDEr drops by only 8 points when only predicted skeletons are used for caption generation compared to the baseline. This initial result shows that skeletons are a promising direction towards unpaired captioning.

3.2.3.2 Human Evaluations

Automatic metrics often have been found not to correlate well with human scores (Kilickaya et al., 2017; Alikhani et al., 2020) and do not fare well when ground truth text is noisy. So we conduct extensive human evaluations where captions for each image are evaluated both in relative preferences and absolute scale (Thapliyal and Soricut, 2020). As mentioned above, we use the T2 test set of 1000 images, each rated by three distinct annotators. The interface of this evaluation is displayed in Figure 3.5. While comparing two models side-by-side, they are randomly assigned ‘A’ or ‘B’ in the interface for each image to avoid any rater bias.


Image	Captions	Please compare caption A to Caption B	Please select individual ratings for each cation
	Caption A: a city from the trails	<input checked="" type="radio"/> A is much better than B <input checked="" type="radio"/> A is better than B <input checked="" type="radio"/> A is slightly better than B <input type="radio"/> A is about the same as B	How does Caption B describe the image? <input type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Acceptable <input type="radio"/> Bad <input type="radio"/> Not enough information
	Caption B: a view of the mountains	<input type="radio"/> B is slightly better than A <input checked="" type="radio"/> B is better than A <input checked="" type="radio"/> B is much better than A	How does Caption B describe the image? <input type="radio"/> Excellent <input type="radio"/> Good <input type="radio"/> Acceptable <input type="radio"/> Bad <input type="radio"/> Not enough information

FIGURE 3.5: Human evaluation interface: We ask raters to: 1) compare the two captions (relative), 2) give ratings for each caption (absolute).

Relative Rating: For each image, we ask the raters to choose the most relevant caption. Comparing Caption A to Caption B, raters can select relative options as shown in the third column in Figure 3.5. *Wins* are the percentage of images where at least 2 out of 3 annotators voted for caption generated with our approach. *Losses* are a percentage of images where at least 2 out of 3 annotators voted for caption generated with the Img2Cap approach. We compute *gains* in this side by side relative evaluation as $Gains_{relative} = Wins - Losses$.

Results: Table 3.6 presents the human ratings for English captions using different skeletons. From this, we observe the following:

- *Dual Staging helps:* Our dual staged models with skeletons (SkeEnc, SkeDec, SkeAE) show gains compared to the improved baseline Img2Cap model. Most notably, it shows that the ‘Nouns & Verbs’ skeletons significantly improve the SkeEncoding model attaining the most significant gain, followed by SkeAE and then SkeDecoding.
- *Subselecting content words helps:* Using the same dual staged SkeEnc model without subselecting content words in the form of iterative refinement does not show any performance improvement, supporting the hypothesis that sub-selecting content skeleton from noisy captions improves the overall caption quality.
- *Cross-lingual skeleton transfer:* Table 3.7 presents our human evaluation scores for captions in other target languages. We observe gains from the skeleton-based approach for 4 out of 5 languages and only a slight loss for the fifth language, showing the effectiveness of cross-lingual transferability of the skeleton words.

3.2.3.3 Cross-modal Discourse Coherence

To understand where the improvements quantified in Table 3.6 come from, we turn to the notion of discourse coherence. Alikhani et al. (2020) introduce multimodal discourse coherence relationships between image-caption pairs. For instance, a caption describing visually

	Baseline caption	magic	peace harbour heaven	view mountain	storm darkness	house nest valley mountain
	property image # apartment for people in a picturesque village	the magic of the colours	the peace of the glorious landscape	the view from the mountains	a dark storm in the darkness	a house nestled in the valley of mountains
	a view from the water	the magic of the lakes	the peace of the river	the view from the mountains	a dark storm on the horizon	the house nestled in the valley of mountains

FIGURE 3.6: Controllability: Effect of guiding the information through the skeleton. As observed, the caption incorporates information from the skeleton that is consistent with the image. For example, we see that peace is incorporated in the second column of the top row while harbor and heaven are not. The relevant skeleton words in other columns guide the captions accordingly.

Approach	Skeleton	Wins	Losses	Gains
SkeEncoding	Nouns & Verbs	39.34	28.33	+11.0
SkeAE	Nouns & Verbs	39.34	32.63	+6.7
SkeDecoding	Nouns & Verbs	34.83	34.53	+0.3
SkeEncoding	Iterative Refinement	19.62	20.52	-1.1

TABLE 3.6: Human evaluation scores of different approaches and skeletons on English (vs the Img2Cap baseline).

Language	Wins	Losses	Gains
French	31.43	29.53	+1.9
Italian	26.13	24.93	+1.2
German	35.23	33.93	+1.3
Spanish	34.03	34.33	-0.3
Hindi	33.13	28.63	+4.5

TABLE 3.7: Human evaluation results for skeleton (form: nouns & verbs, prediction approach: generation) conditioned caption generation for multiple languages.

recognizable aspects of the image, such as ‘people’ or ‘cake’, is annotated using a *Visible* relation; in contrast, a *Meta* relation corresponds to a caption containing details regarding how/when/where the image was captured, such as in ‘warm summer afternoon’, while a *Story* relation implies that the caption describes some potentially non-visible context behind the scene depicted in the image, such as ‘fifth anniversary’.

We hypothesize that our multi-stage approach of skeleton-based IC results in the generation of more captions of *Visible* type, as the intermediate skeleton predictor is trained to predict nouns and verbs from the image. To assess this effect, we train the relation classifier described in Sec. 4 of Alikhani et al. (2020), and obtain discourse relation labels for captions generated on T2-test images by both the baseline Img2Cap and our SkeEncoding models. Table 3.8 (Counts columns) quantifies the shift of relation label distribution towards the *Visible* coherence relation, confirming our hypothesis. We also study the breakdown by coherence relations using the results from our human evaluations on the English captions. Table 3.8 (Human Evals column) reports this breakdown, indicating that, of the 11.01% gains on human evals from Table 3.6, the shift from non-Visible to Visible discourse captions is associated with clear increases in preference from the human raters. This is attributable to the fact that human raters are more likely to prefer captions that are in a *Visible* relation with the image. Therefore the shift towards generating *Visible*-type captions can be positively quantified in terms of human preference.

	Counts			Human Evals
	Baseline	Ours	Change	
<i>Visible</i>	605	640	+5.79%	+10.93%
<i>Meta</i>	245	226	-7.76%	+13.06%
<i>Story</i>	129	108	-16.28%	+10.08%

TABLE 3.8: Analysis of multimodal discourse coherence relations for baseline and our model on T2 dataset. The last column shows the relative human evaluation gains over baseline caption of each type. Other relations with small counts are ignored in the above analysis.

Comparison of SkeEnc and SkeAE on multilingual captions We have discussed the human evaluation scores of the SkeAE model by using *nouns and verbs* as skeletons in Table 3.7 in the main paper. In addition to this, we also conducted human evaluation to compare the SkeEnc model with the *nouns and verbs* skeletons in comparison to the baseline. We present this in Table 3.9. While there are improvements in the three languages, the performance is also hurt in two languages. However, as we see, by comparing the performances in Table 3.7 and Table 3.9, we observe that SkeAE has a clear advantage when leveraging the English caption to improve multilingual captions. This clearly indicates that channeling the prediction of the skeleton words in conjunction with the caption itself enables the model decoder to attend to the previously predicted skeleton words in the same decoder.

Language	Wins	Losses	Gains
French	31.93	31.43	+0.50
Italian	33.13	28.32	+4.81
German	29.43	29.72	-0.30
Spanish	30.53	34.43	-3.90
Hindi	29.93	26.03	+3.90

TABLE 3.9: Human evaluation results on SkeEnc model for skeleton (form: nouns & verbs, prediction approach: generation) conditioned caption generation for multiple languages.

Comparison of Classification and Generation based Skeleton Prediction From a preliminary manual analysis, we observed that the classification based approach to skeleton prediction faces the problem of predicting words that are related but are not likely to co-occur within the same sentence in the caption. This is described in detail in points 1a and 1b of §3.2.2. To validate this observation, we conducted human evaluation of the captions generated from classification and generation based approaches relative to one another. This setup is different from the rest of the experiments in human evaluation in the paper which compare any given model relative to the baseline model. In contrast, this study is to compare the generation and classification approaches with one another. These results are presented in Table 3.10.

The top-8 highest scoring content words are chosen to reduce input noise for the caption generator while improving the recall of concepts. We experimented with different values for this and selected 8 to be an optimal balance between the content in the skeleton words and the noise.

We observe that the generation-based approach has significant gains of +8.91 over the classification-based approach. Most of the prior literature uses the classification-based approach to predict

Approach	Wins	Losses	Gains
Generation	39.14	30.23	+8.91

TABLE 3.10: Human evaluation results of comparison between the generation and classification based approaches

content or bag of concepts to assist caption generation. We hypothesize that this classification-based model helps in end-to-end approaches where the loss from caption generation backpropagates to the classifier model as well. As opposed to this, our model decouples the prediction of the skeleton or concept words that are further used for caption generation. Hence we believe that suppressing the words that do not co-occur is important in the skeleton prediction task, and the generation-based approach addresses this problem.

Absolute Ratings In each human evaluation experiment, we also gathered absolute ratings of each caption in addition to the relative ratings. The relative ratings are described in §3.2.3.2. We also gather absolute rating for each of the 2 captions per image. Each caption is rated as acceptable if at least 2 out of 3 annotators rate it as *acceptable*, *good* or *excellent*. $Gains_{absolute} = Accept_{our_approach} - Accept_{baseline}$. However they are not used in this quantitative analysis. We use them only to validate the ratings such that, for example, an “Excellent” rated caption is not annotated as inferior to a “Bad” rated caption for the same image. These ratings are collected to double-check the results of the relative rating as well.

These scores are presented in Table 3.11. The top part of the table indicates the absolute ratings in terms of Good and OK performance for multilingual captions. The second part of the table shows the same scores when the baseline model is compared with the corresponding model and skeleton combination. Each model, i.e., baseline and the proposed model in each row, are rated individually (not relative to one another). The last two columns indicate the performance shift of the corresponding proposed model with respect to the baseline in each of the Good and OK categories.

Row no.	Language	Good Baseline	Good SkeAE	OK Baseline	OK SkeAE	Gains in Good	Gains in OK
1	French	34.63	35.04	61.36	60.66	+0.40	-0.70
2	Italian	35.14	35.44	60.86	62.56	+0.30	+1.70
3	German	43.64	41.04	67.27	68.07	-2.60	0.80
4	Spanish	48.15	46.55	74.37	74.67	-1.60	+0.30
5	Hindi	59.96	66.17	85.99	87.99	+6.21	+2.00
Row no.	Model	Good Baseline	Good Model	OK Baseline	OK Model	Gains in Good	Gains in OK
6	Unpaired	57.36	55.06	86.48	84.28	-2.30	-2.20
7	SkeEnc (Iterative Refinement)	63.76	62.36	87.89	87.49	-1.40	-0.40
8	Nouns and Verbs (SkeEnc)	66.47	63.66	89.39	88.89	+2.81	+0.50
9	Nouns and Verbs (SkeAE)	51.55	56.66	79.68	83.18	+ 5.01	+3.40

TABLE 3.11: Absolute ratings in percentages in Human Evaluations.

Here are some of the observations from these results:

- *Better results of Dual Staged Approach:* As we can see in the last two rows (rows 8 and 9), our proposed SkeEnc and SkeAE show absolute improvements in both categories. This further demonstrates that the proposed dual staged approach is generating better denoised captions when trained on noisy uncurated alt-text-based captions.

- *Sub-selecting content words is better:* Now that we have seen the improvements with the dual staged approach, we now investigate whether sub-selecting content words is important. For this, we present the comparison between rows 7 and 8. Both these models are dual staged with SkeEnc, i.e., encoding the predicted skeleton in the second stage. The only difference is that row 8 sub-selects all nouns and verbs to predict the skeletons, whereas row 8 includes all the words from the captions to predict the skeletons. Row 8 shows better performance compared to row 7. This means that sub-selecting content words contribute to the caption generation in the second stage.

Img2Ske: Classification based prediction Skeleton prediction is posed as a multilabel classification problem where the prediction of a skeleton word s_i is not conditionally dependent on the prediction of another skeleton word s_j . The encoder part remains the same as the baseline followed by optimization with sigmoid cross entropy between the skeleton words \mathbb{S} and image encoding $z_{\mathbb{I}}$, which is the representation of the image from the encoder.

$$\text{Accuracy, } A = \frac{1}{N} \sum_{i=1}^N \frac{|\mathbb{S}_i \cap \hat{\mathbb{S}}_i|}{|\mathbb{S}_i \cup \hat{\mathbb{S}}_i|} \quad (3.1)$$

The skeleton for the second stage is chosen as the ordered list of top-8 (experimentally selected) high-scoring words after the softmax layer. However, conditional independence of skeleton words with one another ignores the co-occurrences of words capable of composing a sentence or a final caption. For instance, classification predictions are composed of words and their synonyms that are highly correlated like $\{person, man, singer\}$. These words definitely are relevant to an image but do not all necessarily co-occur in a sentence.

Table 3.2 presents the precision, recall, and f-scores of the generation and classification-based approaches for skeleton prediction. These metrics, however, are misleading because they do not account for synonyms or semantic similarity. For example, ‘food’, ‘meal’, ‘lunch’ and ‘dinner’ are all distinct labels while computing these metrics, and predicting one instead of the other get heavily penalized even though the effect on downstream caption quality would be minimal. This issue gets amplified by the fact that CC has a rich vocabulary with words such as electricity ‘pylon’ and ‘tower’ referring to the same concept.

Performance drop for Spanish While we have seen improvements in the performance on multiple languages in human evaluation (Table 3.6), we observed a drop in the preference for Spanish captions when we use skeletons. Given the similarity in word order between Spanish and English compared to Hindi, the lower performance of Spanish is an interesting result indeed. Our speculation for this is probably due to the dialect differences. The translation model we used for Spanish is a mix of ‘Spain Spanish’ and ‘Latin American Spanish’, with Latin American Spanish dominating. The evaluation was done by raters from Spain. The dialects are sufficiently different that it would impact the absolute scores.

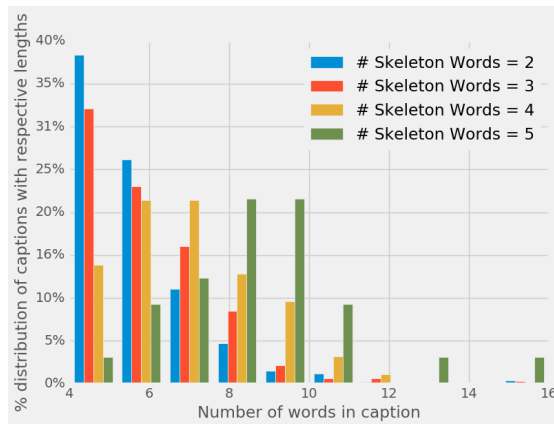


FIGURE 3.7: Quantitative relationship between the number of skeleton words and caption length.


Skeleton Words	valley (1 word)	valley mountain (2 words)	house valley mountain (3 words)	house nest valley mountain (4 words)
	the colours of the valley (5 words)	the green valley of mountains (5 words)	houses of the valley and mountains (6 words)	a house nestled in the valley of mountains (8 words)

FIGURE 3.8: Controllability: Effect of varying the number of words in the skeleton on the generated caption length.

3.2.3.4 Controllability: Qualitative Discussion

The dual-stage modeling decomposition brings forth the advantage of increased interpretability and thereby the ability to use the intermediate stage results to control the final caption. We present aspects of caption controllability by altering the skeleton to explore effects on caption length, informativeness, and gender specificity. This section discusses the utility of this dual staged model for controllability qualitatively. Instead, we present an empirical study only to control gender specificity in two of the languages semi-automatically. We plan to conduct experiments on comparison with other models (Zheng et al., 2019; Chen et al., 2020b) and automatically selecting different but relevant skeleton words in future work.

Effect of length of skeletons on captions: For applications that limit the caption lengths due to UI restrictions, the ability to control the length is important. The length of the skeleton correlates with the number of caption words, as shown in Figure 3.7. For 2 or 3 skeleton words, the percentage of captions monotonically decreases with the number of caption words, with the mode at 4-word captions. Thus, for skeletons of size 2, captions of length 4 are much more frequent than captions of length 6 or 8. For longer skeletons, we see that the mode shifts to the right: with skeletons of size 5, the caption length peaks between 8 and 10 words. Fig 3.8 illustrates this qualitatively.

Effect on gender specificity: Current models often make embarrassing mistakes when generating captions that mention gender. The availability of a skeleton provides a direct handle

for human-in-the-loop correction of such biases at a pre-caption-generation stage. This is more robust compared to caption post-processing, especially for highly inflected languages. To illustrate this, we compare the number of times ‘man’ appears in the captions generated by our baseline versus our dual-stage model after automatically modifying the skeleton (replacing ‘man’ with the gender-neutral word ‘person’ in the skeleton). Over the T2 dataset, the baseline caption generates ‘man’ 13 times and the automatic control mechanism via our model reduces this by 46% (to 7 occurrences) in English. In Hindi, the equivalent of ‘man’ (*Admi*) is generated 10 times, and it is reduced to a gender-neutral word ‘vyakti’ by 70% (to 3 occurrences).

Effect of guiding information through skeleton: The skeleton acts as a knob enabling the model to describe different attributes of the image. Figure 3.6 presents an example of how varying the skeletons for two different images affect their captions. The words highlighted in green are derived from the skeleton, the ones in blue are image-related words.

3.3 N-Local Anchoring from Entities

In this section, we will present our work on using N-Local anchors to drive generation in visual storytelling. We choose these anchors to be entities and their referring expressions. We are enveloped by stories of visual interpretations in our everyday lives. The way we narrate a story often comprises two stages: forming a central mind map of entities and then weaving a story around them. This central representation grounding every sentence in the narrative is what comprises our N-Local anchors. A contributing factor to coherence is not just basing the story on these entities but also, referring to them using appropriate terms to avoid repetition. In this section, we address these two stages of introducing the right entities at seemingly reasonable junctures and referring them coherently in visual storytelling. The building blocks of the central mind map, also known as entity skeleton, are entity chains, including nominal and coreference expressions. This entity skeleton is also represented in different levels of abstractions to compose a generalized frame to weave the story. We build upon an encoder-decoder framework to penalize the model when the decoded story does not adhere to this entity skeleton. We establish a strong baseline for skeleton informed generation and then extend this to have the capability of *multitasking* by predicting the skeleton in addition to generating the story. Finally, we build upon this model and propose a *glocal hierarchical attention model* that attends to the skeleton both at the sentence (local) and the story (global) levels. We observe that our proposed models outperform the baseline in terms of automatic evaluation metric, METEOR. We perform various analyses targeted to evaluate the performance of our task of enforcing the entity skeleton, such as the number and diversity of the entities generated. We also conduct human evaluation from which it is concluded that the visual stories generated by our model are preferred 82% of the time. In addition, we show that our glocal hierarchical attention model improves coherence by introducing more pronouns as required by the presence of nouns.

Storytelling in the age of artificial intelligence is not supposed to be a built-in capability of humans alone. With the advancements in interacting with virtual agents, we are moving towards sharing the ability to narrate creative and coherent stories with machines. The evolution

of storytelling spans from primordial ways of cave paintings and scriptures to contemporary ways of books and movies. In addition, stories are ubiquitously pervasive all around us in digital media. This encompasses multiple modalities, such as visual, audio, and textual narratives. In this work, we address narrating a story from visual input, also known as *visual story telling* (Huang et al., 2016). Generating textual stories from a sequence of images has gained traction very recently (Gonzalez-Rico and Fuentes-Pineda, 2018; Hsu et al., 2018; Kim et al., 2018; Lukin et al., 2018; Peng et al., 2018; Chandu et al., 2019b). Stories can be perceived as revolving around characters (Martin et al., 2018), events/actions (Rishes et al., 2013; Mostafazadeh et al., 2016; Peng et al., 2018), or theme (Gervás et al., 2004). The granularities at which these instances are regulated determine whether the anchoring is *complete* or *partial*. In this work, we choose to anchor our visual stories in characters and their corresponding referring expressions. Emulating a naturally generated story requires equipping machines to learn where to introduce entities, and more importantly, how to refer to them henceforth.

The main task addressed in this section is to introduce entities similar to how humans do and, more importantly, refer them appropriately in subsequent usage. We perform this in two phases: (1) Extraction of n-local anchors, which in this case is entity skeleton extraction, and (2) Anchor-based generation, which is generation informed of the entity skeleton. Here, the anchor or the skeleton is defined as a simple template comprising of the entities and their referring expressions extracted using off-the-shelf NLP tools. The skeletons are what anchors the stories. These skeletons are extracted in three levels of abstraction, comprising of (1) surface form, the skeleton terms in the raw form, (2) nominalized form, that is, the presence of entities in the noun or pronoun form, and (3) abstract form, that is using different notations for based on categories of words from language ontologies. This is delved in more detail in §3.3.2.1. We apply this for the task of visual storytelling, which has both image captions and story sentences in a sequence. Leveraging the captions, the models also inherently learn the association of skeleton to the image captions thereby learning where to talk about which entities in a sequence of images. Once this extraction is performed, we move on to the second phase of incorporating these coreference chains as skeletons to generate a story. This is done in broadly two ways, after conditioning the decoder with the entity skeletons. The first approach is an incremental improvement over the baseline that performs multitasking with an auxiliary goal of predicting the entity skeletons to move the sequences generated from the primary task of story generation closer towards the extracted entities. The second approach is hierarchically attending to the entity skeletons at a local (corresponding to words within a sentence) and global (corresponding to the sentences making up the entire story) levels.

3.3.1 Dataset Description

A dataset that has recently gained traction in the domain of visual storytelling is proposed by Huang et al. (2016). This problem of grounded sequential generation is introduced as a shared task⁴. Formally, the dataset comprises of visual stories or narratives $\mathcal{N} = \{N_1, \dots, N_n\}$. Each story in the dataset consists of a sequence of five story-like images, along with descriptions-in-isolation (DII) and stories-in-sequences (SIS). The descriptions in isolation are isomorphous to image captions. Each story can be formally represented as:

⁴visionandlanguage.net/workshop2018/index.html#challenge

$N_i = \{(I_i^{(1)}, x_i^{(1)}, y_i^{(1)}), \dots, (I_i^{(5)}, x_i^{(5)}, y_i^{(5)})\}$, where $I_i^{(j)}$, $x_i^{(j)}$ and $y_i^{(j)}$ are each image, single sentence in DII and single sentence in SIS respectively, and i refers to the i th example story. SIS and DII are supposed to be associated with each image. However there are about 25% of the images for which DII are absent in the dataset. The corresponding statistics of the dataset are presented in Table 3.12.

	Train	Val	Test
# Stories	40,155	4,990	5,055
# Images	200,775	24,950	25,275
# with no DII	40,876	4,973	5,195

TABLE 3.12: Details of the ViST Dataset

In our modeling approaches as described in §3.3.2, we also need the descriptions in isolation. Hence for the images for which the DII are absent, we use a pre-trained image captioning model to complete the dataset for our use case.

3.3.2 Models Description

The anchors in this section are equivalent to entity skeletons. Our approach of using entity skeletons to generate a coherent visual story is divided into two phases: (1) N-local Anchors Extraction, i.e., entity skeleton extraction, and (2) N-local Anchors Informed Generation. The result of the first step is the anchors, in this case corresponding to n-local anchoring, which results in a sequence of $a_i^{(j)}$ for every j^{th} sentence in the story. Identifying these anchors is a crucial step based on which generation is done in the second step. However, as discussed previously, we use simple off-the-shelf techniques to derive these anchors used in the second step. In this subsection, we first describe three kinds of schema extraction for coreference chains and then proceed towards describing two baselines and two proposed story generation models.

3.3.2.1 Anchor Extraction: N-Local Anchor Representation

The task is to introduce the characters at the right times and refer to them appropriately henceforth. This means that we not only target the head mention of an entity but also cater to the corresponding appropriate coreference expressions. We define these N-Local anchor units or skeleton as a linear chain of entities and their corresponding referring expressions. There could be multiple coreference chains in a long narrative. We associate a story with the entity skeleton that has maximum representation in the five sentences. This means that the skeleton elements need to be present in the majority of the sentences, thus making it the central theme for basing the story on. For simplicity purposes, in case of a tie with respect to the above criterion of the number of mentions, we select the anchor units to be the most frequently occurring coreference chain from among all the chains present in it. In our future work, we plan to extend this capability to cater to multiple skeletons simultaneously. We use off-the-shelf tools to represent these skeletons in three different ways.

Sentences from SIS	Surface	Nominalized	Abstract
The cake was amazing for this event!	None	[0, 0]	None
The bride and groom were so happy.	The bride and groom	[1, 0]	person
They kissed with such passion and force.	They	[1, 1]	person
When their son arrived, he was already sleeping.	their	[1, 1]	person
After the event, I took pictures of the guests.	None	[0, 0]	None
The cake was amazing for this event!	event	[1, 0]	other
The bride and groom were so happy.	None	[0, 0]	None
They kissed with such passion and force.	None	[0, 0]	None
When their son arrived, he was already sleeping.	None	[0, 0]	None
After the event, I took pictures of the guests.	event	[1, 0]	other

TABLE 3.13: Examples of three forms of Entity-Coreference Schema Representation for representing n-local anchors

Anchor Form Representation: For each of the following anchor representations, we first extract the coreference chains from the textual stories that are made up of SIS in the training data. This is done by using version 3.7.0 of Stanford CoreNLP toolkit (Manning et al., 2014). These three ways of representing skeletons are described in detail next.

1. Surface form Coreference Chains: The resulting coreference chains now comprise surface word forms of entities and their corresponding reference expressions. In specific, the N-Local anchor units or skeleton for each story is represented as $\{c_1, \dots, c_5\}$, where c_j is the coreference word in j th sentence. An example of this can be seen in Table 3.13. From the story sentences on the left, there are two entity chains that are extracted corresponding to ‘*the bride and the groom*’ and ‘*event*’. The skeleton word is *None* when there is no word corresponding to that coreference chain in that sentence. The following two columns show the surface form entity skeletons corresponding to the N-Local anchor units for each sentence. Note that there could be multiple such chains extracted for each story due to the number of different entities present in the story. Our goal is to pivot the story on a central mind map, so we select the chain that has the minimum number of *Nones* in the five sentences. Hence in this example, we go ahead with the first skeleton with ‘*the bride and groom*’ to weave the story since the skeleton with ‘*event*’ has a higher number of *Nones*.

2. Nominalized Coreference Chains: The surface form anchor units extracted as described before do not comprise the information of whether it is the head mention of the entity or whether it is referred later. In crude terms, it does not cater to abstracting the properties of the skeleton words from the surface form word itself. The remaining two forms of anchor representations address this issue of abstracting the lexicon from the properties of the word. In order to encode this information explicitly, we disintegrate the bits that correspond to the properties of presence and absence of the entity words and whether the word is present in the noun or the pronoun form. The anchors or skeleton for each story is represented as $\{[h, p]_1, \dots, [h, p]_5\}$. Here, $h \in \{0, 1\}$, is a binary variable indicating if there is a coreference mention, i.e 1 if there is a mention in the skeleton chain and 0 if it is *None*. Similarly, $p \in \{0, 1\}$ is a binary variable indicating that the word is head mention, i.e, the word is in the noun form if it is 0 and pronoun form if it is 1. For instance, in Table 3.13, in sentence 2, the skeleton is represented as [1,0], which means that this sentence has a mention of the skeleton under

consideration and it is in the noun form. Note that we do not use the surface representation of the word itself while we represent the skeleton in this format.

3. Abstract Coreference Chains: As observed from Table 3.13, the anchor units belong to different categories of entities. The first is the raw form in which they appear, and the second is based on the properties of the corresponding anchor units. Instead of disintegrating the properties into nouns and pronouns, another form is to represent them into the abstract categories that they belong to. These categories can be *person*, *object*, *location* etc., This differentiates the order of introduction and references of objects or people in the timeline among the five sentences. We use Wordnet (Miller, 1995) to derive these properties. As depicted in Table 3.13, the entity skeleton corresponding to a coreference chain can be represented with a sequence of ‘*person*’, ‘*other*’ and ‘*None*’.

3.3.2.2 Anchor Informed Generation

In this section, we describe the baseline model used to generate textual stories from visual input. This baseline model is not explicitly anchored in the content derived from an entity and coreference-based N-Local anchor units.

In order to establish a fair comparison, we alter this baseline slightly to establish a second baseline that accesses the anchor units. We then discuss two models that incorporate the entity anchors in various forms in the generation process.

1. Baseline Model: Our baseline model has an encoder-decoder framework that is based on the best performing model in the Visual Story Telling challenge in 2018 (Kim et al., 2018) that attained better scores on human evaluation metrics. The model essentially translates a sequence of images to a story. All of the images are first resized to 224 X 224, and image features are extracted from the penultimate layer of ResNet-152 (He et al., 2016b). These image features act as local features for decoding the sentence corresponding to that image. This sequence of image features is passed through two layers of Bi-LSTMs to obtain the story’s overall context. This contributes to *global* theme of the story. The *local* context for each sentence in the story is incorporated with a skip connection of the local features for that particular image. Finally, the global and local features are concatenated and passed to each time step in the LSTM decoder to generate the story word by word.

For simplicity in formal representation, we use the following notations. Subscript t and superscript τ indicates the t^{th} step or sentence in a story and τ^{th} word within the sentence respectively. I_t , x_t , y_t , represent image, DII, SIS for a particular time step. k_t is the skeleton coreference element for that particular sentence. Here k can take any of the three forms of coreference chains discussed previously, which is word itself (surface form) or a pair of binary digits (nominalization) or noun properties (abstract). Note that k is not used in this baseline model.

The encoder part of the model is represented as the following, which comprises of two steps of deriving the local context features l^t and the hidden state of the t^{th} timestep of the BiLSTM that gives the global context.

$$l_t = ResNet(I_t)$$

$$g_t = Bi-LSTM([l_1, l_2 \dots l_5]_t)$$

The latent representation obtained from this encoder is the *glocal* representation $[l_t \oplus g_t]$, where $[\oplus]$ represents augmentation of the features. This *glocal* vector is used to decode the sentence word by word. The generated words in a sentence from the decoder \hat{w}_t is obtained from each of the words \hat{w}_t^τ that are the outputs that are also conditioned on the generated words so far $\hat{w}_t^{<\tau}$ with τ^{th} word in the sentence being generated at the current step.

$$\hat{w}_t \sim \prod_{\tau} Pr(\hat{w}_t^\tau | \hat{w}_t^{<\tau}, l_t, g_t) \quad (3.2)$$

The baseline model is depicted in the right portion of the Figure 3.9.

2. Skeleton Informed Baseline Model: We need to make a note here that though the above baseline is the best performing model in the task, it does not take into account the explicit mentions of the entities as a skeleton to weave the story on. Similarly, it does not make use of the DII for the images. We explore how to make better use of these DII to extract the entity skeletons. Hence to establish a fair comparison with our proposed approaches we condition the decoder on not only the glocal features and the words generated so far, but also the surface form of the words.

$$\hat{w}_t \sim \prod_{\tau} Pr(\hat{w}_t^\tau | \hat{w}_t^{<\tau}, l_t, g_t, k_t) \quad (3.3)$$

In specific the features that are given to the decoder now have $[l_t, g_t, k_t]$. The skeleton information is provided to every time step in the decoder.

3. Multitask Story Generation Model (MTG): Incorporating the entity skeleton information directly in the decoder might affect the language model of the decoder. Hence we take an alternate approach that incrementally improves upon the first baseline model to enable it to perform two tasks. Instead of augmenting the model with skeleton information, we enable it to predict the skeleton and penalize it accordingly. The main task here is the generation of the story itself, and the auxiliary task is the prediction of the entity skeleton word per time step. Each of these tasks is optimized using cross entropy loss. The loss for generation of the story is L_1 and the loss to predict the skeleton of the model is L_2 . However, we do not want to penalize the model equally for both the participating tasks and weigh them by a factor α as much as to affect the language model of the decoder. We experimented with different weighting factors for α , which are presented in Table 3.14.

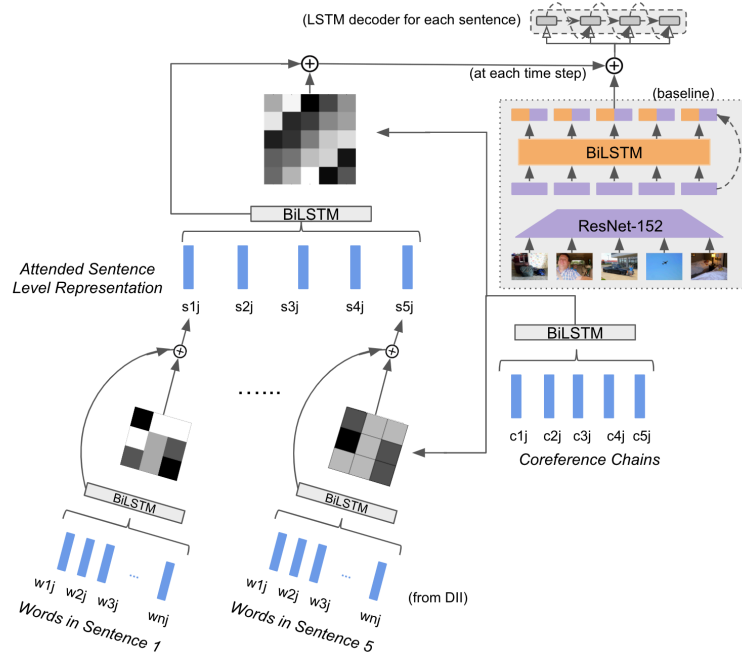


FIGURE 3.9: Architecture of Glocal Hierarchical Attention on Entity Anchors with coreference chains to perform Visual Storytelling

$$\sum_{I_t, x_t, y_t \in \mathbb{S}} \alpha \mathbf{L}_1(I_t, y_t) + (1 - \alpha) \mathbf{L}_2(I_t, y_t, k_t)$$

Note that we do not use k as a part of the encoder even in this model but only use them to penalize the model when the decoded sentence does not contain a skeleton similar to k .

4. Glocal Hierarchical Attention: Enabling the model to predict the entity skeleton equips it to model the sentences around the entities, thereby weaving the stories around the skeleton. However, this multitasking model does not explicitly capture the relationship or focus on the words within a sentence or across the five sentences with respect to the skeleton in consideration. Hence, we went one step further to identify the correlation between the coreference skeleton with different levels, including within a sentence (i.e., at word level) and across sentences (i.e., at sentence level). We use an attention mechanism to represent these correlations.

We propose two stages of attention to capture this information:

1. *Local Attention:* attending to the words in captions (w_t^T from x_t) with respect to the entity skeletons k_t .
2. *Global Attention:* attending to the sentences in the story derived from the local attention for each sentence.

Figure 3.9 depicts the entire glocal hierarchical attention model with the encoder-decoder framework on the right and the two stages of attention on the left. The attention is performed






Models						Phenomena
SIS	we went to the stadium early to eat and sight see before the game .	the view was incredible . you could see the entire city .	we got to our seats , and could n't believe how close to the field they were .	we could see all the action .	once the national anthem was sung , and the first pitch was thrown , the excitement began . it was a great game !	
Baseline Model	the city was a great place to visit .	i had a great time .	there were many people there .	we got to see a lot of cool things .	it was a lot of fun .	- Characters in the story are mentioned as "many people" instead of "we" (sentence 3).
Glocal Hierarchical Attention Model	we saw the building was packed .	i was excited to see my favorite team .	we were all excited to see the game .	we all got together to watch .	it was a great game .	+ characters ('we' and 'it') were introduced at the right time + Important entities were mentioned (building, game)

FIGURE 3.10: Qualitative Analysis

on textual modality corresponding to DII (x_t) and hence can be perceived as translating DII to SIS. As observed in Table 3.12, DIIs are absent for about 25% of the data. We use an image captioning model pretrained on ImageNet data (Russakovsky et al., 2015b). These image captions are substituted in the place of missing DII.

Local Attention: The first level of attention, i.e., the *local attention* measures the correlation between words in each sentence to the coreference skeleton words. There are five sentences in each story corresponding to five images. Since we use the skeleton words as they appear to attend to the words in DII, we use the surface form notation in this model. As we have seen, the surface form skeleton is represented as $C = \{c_1, c_2, \dots, c_5\}$. The vocabulary of these surface form skeleton words is limited to 50 words in the implementation. The surface skeleton form C is passed through a Bi-LSTM, resulting in a hidden state H_k which is of 1024 dimensions. This hidden state is used to perform attention on the input words of DII for each image. Note here that the skeleton words for coreference chains are extracted from SIS (i.e., from $\{y_1, y_2, \dots, y_5\}$), from which the hidden state is extracted, which is used to perform attention on the individual captions (DII i.e., $\{x_1, x_2, \dots, x_5\}$).

The skeleton remains the same for all the sentences. The skeleton form is passed through a Bi-LSTM resulting in $H_k \in \mathbb{R}^{k \times 2h}$, where hidden dimension of the Bi-LSTM is h . Each x in the story (with n words in a batch) is passed through a Bi-LSTM with a hidden dimension of h , resulting in $H_w \in \mathbb{R}^{5 \times n \times 2h}$. This then undergoes a non-linear transformation.

Attention map for the word level is obtained by performing a batch matrix multiplication (represented by \otimes) between the hidden states of the words in a sentence and the hidden states of the entity skeleton. In order to scale the numbers in probability terms, we apply a softmax across the words of the sentence. Essentially, this indicates the contribution of each word in the sentence towards the entity skeleton that is present as a query in attention. This is the *local attention* $A_w \in \mathbb{R}^{5 \times n \times k}$ pertaining to a sentence in the story. Mathematically, equation 3.4 depicts the calculation of *local attention*.

$$A_w = \text{softmax}(H_w \otimes H_k)$$

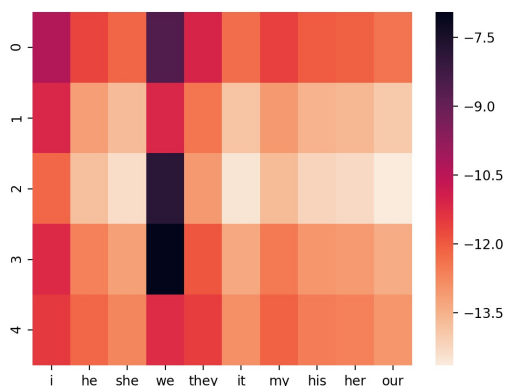


FIGURE 3.11: Visualization of the hierarchically attended representation of the skeleton for story in Figure 3.10

Glocal Attention: We then perform *global attention*, which is at the whole story level. In other words, this attention evaluates the contribution of different sentences in the story towards responding to the extracted entity skeleton. Instead of considering the sentence representation as the output of passing the words as is through a Bi-LSTM, we leverage the already attended local attention (which is at a sentence level) to perform the global attention. Hence it is the combination of global and local attention, and thereby we perform *glocal hierarchical attention*.

For this, each sentence’s locally attended representation is augmented with the output of the Bi-LSTM that takes in DII. The attended representation for each of the k words is concatenated and projected through a linear layer into 256 dimensions (P_w). This goes in as sentence representation for each of the s_{ij} (where i is the index of the sentence in the story and j corresponds to the story example) as shown in Figure 3.9. The word representations at each time step are obtained by augmenting the corresponding vectors from H_w and P_w . These form our new sentence embeddings. These sentence embeddings are again passed through a Bi-LSTM to get a sentence level representation. This process is done for each sentence in the story (which are the replications as shown in the left portion of Figure 3.9). This results in a latent representation of the story $H_s \in \mathbb{R}^{5 \times 2h}$. Along the same lines of local attention, we now compute story level hierarchical global attention to result in $A_s \in \mathbb{R}^{5 \times k}$. This is shown in Equation 3.4 where $[,]$ indicates augmentation of corresponding vectors.

$$A_s = softmax([H_w, P_w] \otimes H_k)$$

The attended vectors from A_w and A_s of size nk and k respectively are concatenated in each sentence step in the decoder from the baseline model. This is shown in the top right corner of Figure 3.9 (although the Figure depicts concatenation for single time step).

The various methods discussed above are presented in their condensed anchoring methods in Figure 3.12.

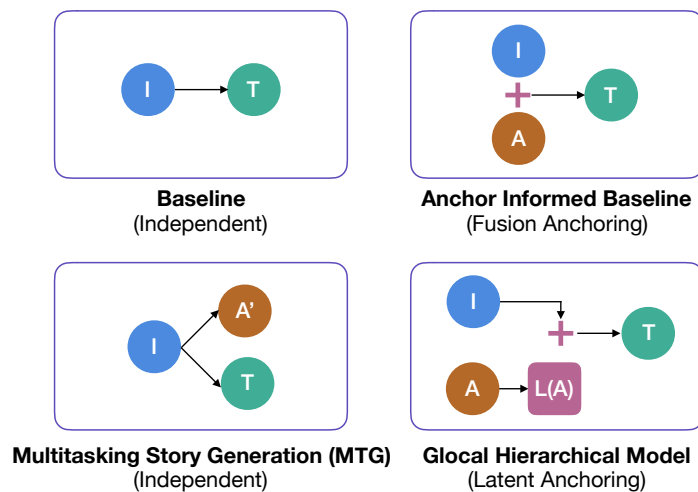


FIGURE 3.12: Comparison of the content-anchoring based on multimodal interactions for modeling entities and coreferences

Models	Entity Skeleton Form	METEOR	Distance	Avg # distinct entities
Baseline	None	27.93	1.02	0.4971
+ Entity Skeletons	Surface	27.66	1.02	0.5014
MTG ($\alpha(0.5)$)	Surface	27.44	1.02	0.9554
MTG ($\alpha(0.4)$)	Surface	27.59	1.02	1.1013
MTG ($\alpha(0.2)$)	Surface	27.54	1.01	0.9989
MTG ($\alpha(0.5)$)	Nominalization	30.52	1.12	0.5545
MTG ($\alpha(0.5)$)	Abstract	27.67	1.01	0.5115
Glocal Attention	Surface	28.93	1.01	0.8963

TABLE 3.14: Automatic Evaluation of Story Generation Models

Hyperparameter setup: Learning rate of 0.001 is used with a batch size of 64. The word embedding dimension is 256 and the image features contributing towards the local representation is 1024. The hidden size of the Bi-LSTM is 1024 which is the dimension of the global vectors. The attention map features are of dimension size of 256. We selected the 50 most frequently occurring coreference words to set the vocabulary of k for these experiments i.e, $size(k) = 50$.

3.3.3 Experiments and Results

This section presents the quantitative and the qualitative results for the four models discussed in the previous section.

3.3.3.1 Quantitative Analysis

We perform automatic evaluation with METEOR score for generation. The results are shown in Table 3.14. However, our main target is to verify whether the story adheres to the provided entity anchor units. Hence we attempt to perform this evaluation with a different scoring

mechanism. We extract anchor units which in this case are entity skeletons from the generated stories in the same procedure as performed on the training stories. With respect to the ground truth stories, a binary vector of length five is constructed based on whether the entity skeleton word is present or not in that sentence. Euclidean distance between these binary vectors skeletons of the original and the generated stories is used to validate this aspect of generation. Table 3.14 presents the results of our models. As we can see, the Euclidean Distance is not very different in each of the cases. However, we observe that the multitasking approach (MTG) performs better with the nominalized form of entity skeletons than the baselines and other forms of entity skeleton representations. The *glocal* model described performs attention on the surface words only, and hence the experiment includes only this configuration. We observe that the *glocal* attention model outperforms the baseline model. However, there is a scope for improvement when the attention mechanism is performed on nominalized skeleton representation, which we leave for future work.

These automatic metrics do not sufficiently capture the number or the diversity of the entities introduced in the generated stories. We calculated the percentages of the nouns and pronouns in the ground truth and the generated stories for the test data to analyze the number of entities. Figure 3.13 presents these percentages for the ground truth stories, generated stories from baseline, MTG with nominalized skeleton representation, and the *Glocal* attention model. As we can see in the nouns section, the baseline model seemed to have over-generated nouns compared to both of our proposed models. While our MTG model also has over-generated the nouns, our *glocal* attention model has generated fewer nouns compared to the ground truth. However, this is still the closest to the number of nouns in the ground truth stories. Generating a high number of nouns does not ensure coherence as much as generating an appropriate number of relevant pronouns. This is observed in the second section in the graph. While the MTG model generated a higher number of pronouns in comparison to the baseline, the *glocal* attention model seemed to have generated an even higher percentage of pronouns. Despite this over-generation, the *glocal* attention model is the closest to the number of pronouns in the ground truth stories. Interestingly, the MTG and *glocal* attention models seem to have opposite trends in the generation of nouns and pronouns. We plan on investigating this further in our future work. Coming to the diversity of the entities generated by the stories, we calculate the average number of distinct entities present per story for each model. These numbers are shown in the last column of Table 3.14. This number for the ground truth test stories is 0.7944. As we can see, the average number of distinct entities is comparatively high for the MTG model. However, this number is closer to that of the ground truth for the *glocal* attention model, assuring sufficient diversity in the entity chains generated by this model. We would like to make a note here that though the MTG model with nominalized representation is performing better in terms of METEOR score, our analysis shows promisingly better performance of the *Glocal* attention model with respect to both the number and diversity of the entities generated.

3.3.3.2 Qualitative Analysis

Figure 3.10 presents an image sequence for a story along with the corresponding ground truth (SIS) and the generated stories. The positive and the negative phenomena observed are presented in the last column. The *Glocal* Hierarchical Attention Model is able to capture the

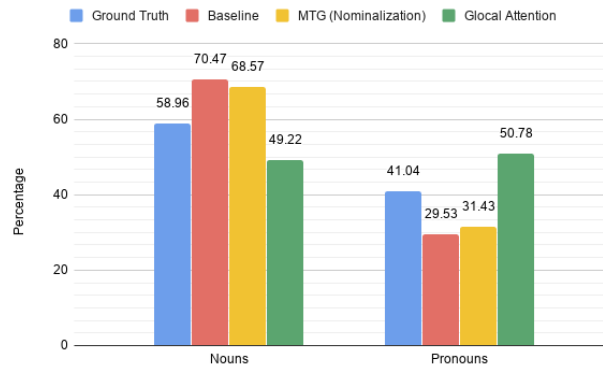


FIGURE 3.13: Percentage of Entities in the form of Nouns and Pronouns in the generated stories

skeleton words right in comparison to the baseline model. For instance, the words ‘we’ and ‘it’ are generated in sentences 1, 3, 4, and 5 with the glocal attention model, whereas these entity skeleton words are generated only in sentences 4 and 5 in the baseline model. Since there are multiple occurrences of entities that are connected, the story might present stronger coherence in the case of the glocal attention model. In addition, the entity skeleton could be boosting the model to also generate other relevant words based on images such as ‘building’ and ‘game’. The visualization of the corresponding hierarchical glocal attention map that is fed into the decoder is presented in Figure 3.11. Darker color indicates higher attention on those words. The rows in the visualization depict the sentence indices, and the columns indicate a few of the frequently occurring entity skeleton chains. The scores are not normalized as probability distributions since the figure does not present all 50 entity skeletons (instead of only the top 10 frequently occurring ones). As we can see, there is a higher weight in the grids pertaining to ‘we’ for the first, third, and fourth sentences.

Human Evaluation: We conduct human evaluation in the form of preference testing. Twenty stories were randomly sampled, and we asked five subjects the following preference questions ‘preference of the story narrative from the images’. Our *glocal hierarchical attention model* is preferred 82% of the times compared to the baseline model and 64% of the times in comparison to the MTG model with nominalized representation. We also asked them a follow-up question of their opinion on the usage of pronouns since that is the task we were focusing on. From the answers, we conclude that our hypothesis of the usage of pronouns instead of third-party nouns narrates a more involved story. Therefore, this provides an opportunity margin for improving story generation.

So far, we have seen how N-Local anchor units based on entity skeletons can provide fine-grained guidance towards anchoring the content of the story in the entities involved in the stories. What if the anchors are not explicitly available? Can they be leveraged from the surrounding contexts? In what kind of narratives do surrounding contexts help anchor the content more? To investigate these questions, let us look into anchoring with implicit contexts in the following section.

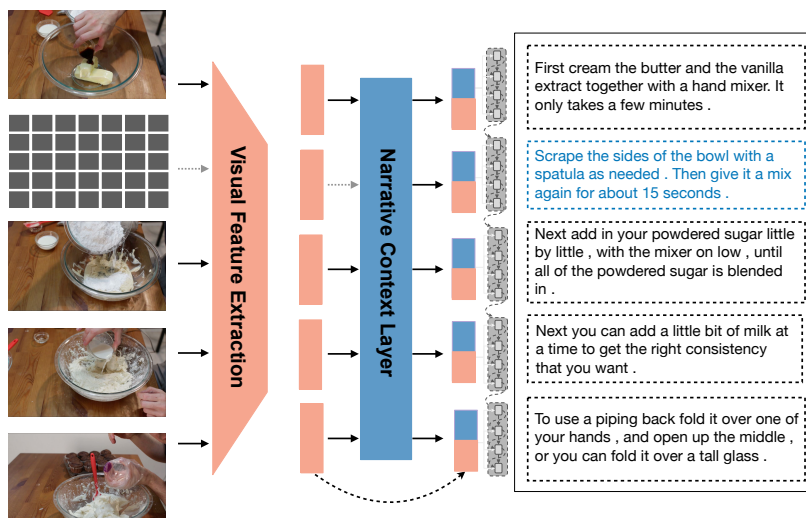


FIGURE 3.14: Overview of infilling in visual procedures. Image in the second step is masked while the model generates the corresponding textual description from surrounding context.

3.4 N-Local Anchoring from Implicit Contexts

Generating long form narratives such as stories and procedures from multiple modalities has been a long-standing dream for artificial intelligence. In this regard, there is often a crucial subtext that is derived from the surrounding contexts. The general seq2seq training methods render the models shorthanded while attempting to bridge the gap between these neighboring contexts. In this paper, we tackle this problem by using *infilling* techniques involving the prediction of missing steps in a narrative while generating textual descriptions from a sequence of images. We also present a new large scale *visual procedure telling* (ViPT) dataset with a total of 46,200 procedures and around 340k pairwise images and textual descriptions that is rich in such contextual dependencies. Generating steps using infilling technique demonstrates the effectiveness in visual procedures with more coherent texts. We conclusively show a METEOR score of 27.51 on procedures which is higher than the state-of-the-art on visual storytelling. We also demonstrate the effects of interposing new text with missing images during inference. The code and the dataset will be publicly available at <https://visual-narratives.github.io/Visual-Narratives>.

Humans process information from their surrounding contexts from multiple modalities. These situated contexts are often derived from a modality (source) and expressed in another modality (target). Recent advances have seen a surge of interest in vision and language as source and target modalities respectively. One such widely studied task is image captioning (Hosain et al., 2019; Liu et al., 2019) which provides a textual description T given an image I . In contrast, visual storytelling (Huang et al., 2016) is the task of generating a sequence of textual descriptions $(\{T_1, T_2, \dots, T_n\})$ from a sequence of images $(\{I_1, I_2, \dots, I_n\})$. This sequential context is the differentiating factor in the generation of visual narratives in comparison to image captioning in isolation. This long form generation comprises a coherent sequence of multiple sentences.

A fundamental incongruity between how humans process information from multiple modalities and how we teach machines to do the same is that humans are capable of bridging the information gap from surrounding contexts. Our training procedures do not take care of accommodating the same ability in a supervised learning paradigm. Traditionally, the problem of missing context in long text generation is addressed using additional input such as entities (Dong et al., 2019c), actions (Fan et al., 2019b), latent templates, external knowledge etc.. These are explicit methods to inject content during generation. In contrast, in the spirit of simplicity, we propose infilling techniques to implicitly interpolate the gap between surrounding contexts from a stream of images. The training procedure incorporates masked contexts with the objective of a masked span prediction. We focus on two kinds of visual narratives namely, stories and procedures. We curated a large scale *ViPT* dataset with pairwise image and text descriptions comprising of 46k procedures and 340k images. The percentage of unique words in each step in comparison to the rest of the recipe is about 60% for ViST and 39% for ViPT. This implies that overlapping contexts are predominant in procedures than stories datasets. This is usually because stories are more creative and diverse while procedures are in-domain. For both these reasons, we hypothesize that infilling technique is more effective in scenarios where it can leverage the vast context from the surrounding information to fill the missing pieces. To this end, we present our infilling-based model to perform visual narrative generation and compare its effects on visual stories and procedures. The overview of the infilling-based training procedure is presented in Figure 3.14. We conclusively observe that it is more effective in procedural texts with stronger contextual dependencies. We also present the effects of infilling during training and inference phases and observe that infilling shows benefits during inference as well. Similarly, infilling-based techniques are also capable of generating longer sentences. Interpolating contexts to generate narrative descriptions has potential applications in fields such as digital education (Hollingshead, 2018), social media content (Gella et al., 2018), augmented reality (Dudley et al., 2018), video games (Kurihara et al., 2019; Ammanabrolu et al., 2019a), etc.. The main contributions of this paper are:

- We present a Visual Procedure Telling (ViPT) dataset similar to the Visual Storytelling (ViST) dataset with 46k procedures on various domains.
- We demonstrate the efficacy of our visual infilling technique on narratives that have stronger contextual dependencies on the rest of the sentences.

Dataset	ViST	Visual Procedure Telling (ViPT)									
Categories	stories	recipes	crafts	outdoors	lifestyle	technology	styling	fitness	hobbies	pets	misc
#narratives	50,136	34,138	660	1,831	1,824	1,660	1,585	911	1,701	858	1,032
#images or steps	209,651	203,519	8,658	20,526	20,959	19,221	18,112	9,935	19,145	9,599	11,853
avg #steps	5.00	5.96	13.12	11.21	11.49	11.57	11.42	10.90	11.25	11.18	11.48
avg #words/step	11.35	79.19	47.99	35.52	32.58	27.90	17.31	17.54	17.54	17.24	57.45

TABLE 3.15: Details of the ViST and *Visual Procedure Telling* Dataset broken down into 10 categories

3.4.1 Dataset Description

While there are several types of narratives, such as literary, factual, and persuasive, this paper looks into stories and procedures. This section describes our new ViPT dataset and highlights the differences with ViST.

Procedures vs Stories: Long form narratives are often characterized by three crucial properties: content, structure, and surface form realization (Gatt and Krahmer, 2018b). Narrative properties such as content and structure in these forms are sufficiently contrastive between stories and procedures. Content in stories includes characters and events, while procedures include ingredients, materials, and actions. Coming to the structure, stories typically start by setting a scene and the era, followed by characterizing the participants and culminating with a solution if an obstacle is encountered. In contrast, a procedural text is often goal-oriented and thereby typically begins by listing the ingredients/materials needed, followed by a step-by-step description to arrive at the final goal. While stories can be metaphoric, sarcastic, and humorous in surface realization, the sentences in procedures are often in imperative or instructional tone.

2. Data Collection Process: We manually examined around 10 blogging websites with various user-written text on several *how-to* activities. Among these, we found that *snapguide* and *instructables* are consistent in the form of pairs of textual descriptions along with their images. We are going to release the scripts used to collect this data as well as preprocess them. We removed all the procedures in which at least one image in each step is absent. Once all this preprocessing is done, the data contained the following categories in both websites. These categories are based on the tags given by the bloggers to the articles they have written from among the categories that each website offers. These categories for each of these websites are:

- *snapguide*: recipes, games-tricks, sports-fitness, gardening, style, lifestyle, outdoors, beauty, arts-crafts, home, music, photography, pets, automotive, technology
- *instructables*: crafts, cooking, teachers, circuits, living, workshop, outside

In union, they are a total of 18 categories. We manually examined a few procedures in each of the categories and regrouped them into 10 broad categories that are presented in Table 3.15. A list of URLs corresponding to the data is submitted along with the paper.

Visualization of topics: Each of the categories in our Visual Procedure Telling (ViPT) are analyzed for the topics present in them. To get a more detailed understanding of these topics in the dataset, we hosted the topic visualizations here: visual-narratives.github.io/Visual-Narratives/.

category	snapguide	instructables
recipes	desserts, food	cooking
crafts	arts-crafts	craft
outdoors	outdoors, gardening	outside
lifestyle	lifestyle, home	living
technology	technology, automotive	circuits
styling	style, beauty	
fitness	sports-fitness	
hobbies	music, photography	
pets	pets	
misc	games-tricks games-tricks	teachers, workshop

TABLE 3.16: Regrouping the categories in ViPT dataset

ViPT dataset: Though stories have the potential to exhibit the properties listed above, it is challenging to observe them in the ViST dataset (Huang et al., 2016) owing to the shorter sequence lengths. The extent to which adjacent groups of sentences have overlapping contexts is high in procedures as compared to stories. We had previously gathered cooking recipes to experimentally demonstrate a scaffolding technique to improve structure in long form narratives (Chandu et al., 2019a). We extend this work to gather procedures or ‘how-to’ articles that have step-by-step instructions along with an associated pairwise image to each step in several domains. To facilitate multi-domain research with stronger interleaved contexts between surrounding steps, we present a large scale *visual procedure telling* dataset with 46k procedures comprising of 340k pairwise images and textual descriptions. It is carefully curated from several *how-to* blogging websites. Our dataset comprises pairwise images and textual descriptions of the corresponding images, typically describing a step in a procedure. This means that each description of the step is tethered to an image. This makes it a *visual narrative telling* task. We categorized the dataset into 10 distinct domains, including recipes, crafts, outdoors, lifestyle, technology, styling, fitness, hobbies, pets, and miscellaneous. The category-wise details of the dataset are presented in Table 3.15. As we can observe, the dataset is dominated by cooking recipes which are relatively of similar sizes as that of ViST compared to the rest of the domains.

Differences between ViPT and ViST datasets: As observed in Table 3.15, the average number of steps in ViPT is higher than that of ViST. However, the average number of steps in recipes and stories is similar, 5.96 and 5.00, respectively. The average number of words per step in ViPT is also much higher, thereby presenting a more challenging long form text generation task. Despite the average number of steps being similar, the average length of each step i.e., the number of words per step in cooking recipes, is about seven times that of stories. Typically, each step in the ViPT dataset comprises of multiple sentences that is indicative of the corresponding image. This is as opposed to ViST dataset, which has a single sentence per step. These long sequences also present a case for dealing with larger vocabularies as well. The recipes category alone has a vocabulary of 109k tokens while the same for stories is 25k. We also compared the diversity in vocabulary of each step by computing the average percentage

of unique words in a step with respect to the rest of the narrative. While this number is a high 60% for ViST, it is 39% for ViPT. This means that about 40% of the words in each step in ViST overlap with the rest of the story. This could be owed to the way the dataset is gathered by asking the annotators to pick a sequence of images that are likely to make a coherent story and then describe these images in sequence. While the stories-in-sequences sufficiently distinguish themselves from descriptions-in-isolation, the overlapping contexts are not high compared to procedures. The overlapping contexts for procedures is about 61%. This reveals the stronger cohesive and overlapping contexts in the ViPT dataset compared to the ViST datasets. These overlapping contexts motivate the idea of generating a sentence by bridging the contexts from surrounding sentences. Hence it forms a suitable testbed to learn interpolation from surrounding contexts with infilling technique.

Dataset	Stories				Recipes			
	XE	V-Infill	V-InfillR	INet	XE	V-Infill	V-InfillR	INet
BLEU-1	62.05	61.58	61.84	63.31	28.61	29.73	28.61	25.10
BLEU-2	38.31	37.27	37.81	39.60	16.89	17.50	17.01	13.36
BLEU-3	22.68	21.70	22.42	23.62	10.50	10.83	10.59	6.51
BLEU-4	13.74	12.96	13.69	14.30	5.68	5.81	5.71	3.60
METEOR	35.01	34.53	35.08	35.57	26.72	27.26	27.51	25.62
ROUGE_L	29.66	29.12	29.65	30.14	21.64	22.02	18.66	20.43

TABLE 3.17: Performance of different models on stories (from ViST) and recipes (from ViPT) datasets

Infill Index	0		1		2		3		4		5	
	XE	V-Infill	XE	V-Infill	XE	V-Infill	XE	V-Infill	XE	V-Infill	XE	V-Infill
BLEU-1	20.9	29.7	22.8	29.8	23.5	29.9	24.4	30.4	25.5	31.0	26.4	31.5
BLEU-2	12.5	18.0	13.2	17.6	13.6	17.5	14.2	17.8	14.9	18.2	15.4	18.6
BLEU-3	7.9	11.1	8.2	10.7	8.4	10.8	8.8	10.9	9.2	11.1	9.6	11.4
BLEU-4	4.2	5.8	4.2	5.6	4.4	5.6	4.7	5.7	4.9	5.8	5.1	6.0
METEOR	27.6	27.8	26.4	27.1	26.0	26.9	26.3	27.1	26.6	27.2	26.8	27.4
ROUGE_L	20.9	22.4	20.3	21.8	20.6	21.8	21.0	21.9	21.3	22.0	21.5	22.1

TABLE 3.18: Performance of infilling during inference for recipes in Visual Procedure Telling

Infill Index	0		1		2		3		4	
	XE	V-Infill	XE	V-Infill	XE	V-Infill	XE	V-Infill	XE	V-Infill
BLEU-1	60.9	63.0	60.8	62.0	60.3	61.9	60.5	62.2	61.8	63.3
BLEU-2	37.0	39.5	36.9	38.6	37.0	38.4	37.0	38.7	38.1	39.6
BLEU-3	21.7	23.7	21.6	23.1	21.8	22.9	21.8	23.2	22.5	23.7
BLEU-4	13.1	14.4	13.1	14.1	13.2	13.9	13.3	14.3	13.8	14.5
METEOR	34.9	35.4	34.8	35.1	35.2	35.2	35.1	35.3	35.2	35.5
ROUGE_L	29.3	30.2	29.2	29.9	29.1	30.0	29.2	30.0	29.5	30.3

TABLE 3.19: Performance of infilling during inference for Visual Story Telling

3.4.2 Models Description

This section describes the baseline model and the infilling techniques adopted on top of it.

We present infilling-based techniques for learning missing visual contexts to generate narrative text from a sequence of images. As the ViST and recipes category in ViPT are of comparable

sizes (both in terms of data size and the average number of steps per instance), we perform comparative experimentation on these two categories. We leave experimenting with all the domains for our future work, especially learning from one domain to generate the sequences in other domains. For our ViPT category, we use 80% for training, 10% for validation, and 10% for testing. The stories are composed of 5 steps, and the cooking recipes are truncated to 5 steps to perform a fair comparison of the effect of the index being infilled. An overview of infilling-based training is depicted in Figure 3.14. The underlying encoding and decoding stages are described here.

Encoding: Models 1, 2, and 3 here show different variants of encoding with and without infilling. Model 4 is the state-of-the-art model for generating stories on ViST. Note that the encoding part of the missing contexts varies between these models while the decoding strategy remains the same to compare (i) the performance of encoding masked contexts as opposed to not masking, and (ii) the performance of masked span prediction between stories and procedures.

1. XE (baseline): We choose a strong performing baseline model based on sequence to sequence modeling with cross entropy (XE) loss inspired from Wang et al. (2018). It is a CNN-RNN architecture. The visual features are extracted from the penultimate layer of ResNet-152 by passing the resized images ($\{I_1, I_2, \dots, I_n\}$) of size 224 X 224. These represent the image specific local features ($\{l_1, l_2, \dots, l_n\}$). These features are then passed through a bidirectional GRU layer to attain narrative level global features ($\{g_1, g_2, \dots, g_n\}$) constituting the narrative context layer in Figure 3.14.

2. V-Infill: We introduce an infilling indicator function on the underlying XE model by randomly sampling an infilling index (in_{idx}). This is used to construct the final infilled local features as follows.

$$l_k(\forall k, s.t. 0 < k \leq n) = \begin{cases} zero_tensor & \text{if } k=in_{idx} \\ l_k & \text{otherwise} \end{cases}$$

Other than the sampled in_{idx} , the rest of the local features for other indices remain the same. The local features for in_{idx} are all masked to a zero tensor. The dropout of an entire set of local features from an image forces the model to learn to bridge the context from the left and the right images of in_{idx} . The model is optimized to predict the rest of the steps where images are present along with the *masked span prediction*. In this way, the infilling mechanism encourages our underlying seq2seq model to learn the local representation of the missing context from global contextual features in the narrative context layer.

3. V-InfillR: This model varies the Rates in which local features are masked as training proceeds based on the indicator function above in the V-Infill model. Scheduling the number of missing features itself is a hyperparameter, and we used the following setting. In the first quarter of training epochs, none are masked, then increasing it to 1 local feature for the next

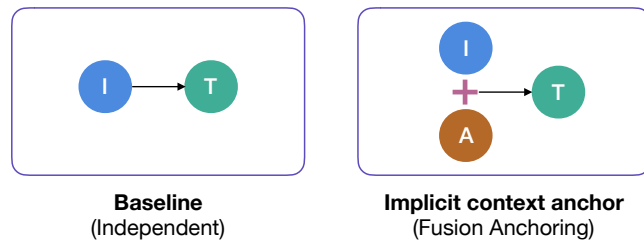


FIGURE 3.15: Comparison of the content-anchoring based on multimodal interactions in infilling

quarter and leaving it at 2 for the last two quarters. This is similar to the settings observed in INet model. We have experimented with other settings of scheduling, but this one performed better than the others.

As mentioned earlier, the encoding of the local features changes based on the infilling technique being used in each of the above strategies. As we can see, the contribution of the global features to reconstruct the missing local context is intuitively expected to perform well in the case of narratives with overlapping contexts. Hence, we hypothesize that the infilling technique that interpolates between steps that constitute words or phrases that are similar to those of the surrounding steps benefit from this technique. A ‘*how-to*’ style of narrative explaining a procedure is more in-domain as compared to the stories and hence hypothesize that our infilling-based encoding approaches perform relatively better on procedures. We then use the encoded representation to decode each step of the procedure or story. The decoding strategy is explained next which is the same in all three of the aforementioned models.

Decoding: In all the above models, g_k are fed into a GRU decoder to predict each word (\hat{w}_t) of the step (k). The same is done for generating each step in the five steps. In the infilling methods, the decoding strategy is agnostic to the missing context in the local features. The global features that bridge the contexts in the encoding are used directly as input to the decoder. In other words, the network remains the same once the global features are predicted. We perform beam search with a beam size of 3 during inference. Here τ is the number of words in each step, and t is the current time step.

$$\hat{w}_t \sim \prod_{\tau} Pr(\hat{w}_t^{\tau} | \hat{w}_t^{<\tau}, g_k)$$

4. INet: We re-implemented the model achieving the state-of-the-art results (Hu et al., 2020b) on the visual storytelling dataset. Additionally, they use a relational embedding layer that captures relations across spatio-temporal sub-spaces. Our replication of their model is close to the scores reported in their paper, though not exact. Our re-implementation achieved a 35.5 METEOR and 63.3 BLEU-1 in comparison to the scores reported in their paper which are 35.6 and 64.4.

The comparison between the baseline and the implicit context model with infilling is depicted in Figure 3.15.

Hyperparameter Setup: We use a GRU with a hidden dimension of 256 for the encoder and 512 for the decoder. The word embedding dimension is 512. The learning rate is $4e-4$ optimized with Adam and smoothing of $1e-8$. We use a dropout of 0.2 and momentum of 0.9 with a gradient clipping of 10. When experimented with a transformer-based encoder and autoregressive decoding, the performance is comparatively lesser, and hence we proceed with a GRU-based model. Based on the average number of steps in recipes from Table 3.15 which is 5.96, we truncate the recipes to 6 steps.

3.4.3 Experiments and Results







	0	1	2	3	4	5		
								
GT	There are only a few components necessary to sear meat and make a pan sauce.	Trim the fat around the meat until almost none is left .	Dry the meat by blotting it with paper towels .	Heat a little oil on a stainless steel or cast iron skillet on high heat and wait for the oil to start smoking .	After 3-5 minutes pass , use a spatula to check the meat.	Once the meat is to your desired temperature , rest on a plate and cover with foil , 5-10 minutes .	Analysis	
XE	0	Cut the chicken into small pieces .	The first thing you will need to do is put the meat in the pan .	Heat the oil in a pan and add the olive oil .	Add the salt and pepper to the pan .	Once the chicken is cooked , remove the chicken from the oven and let it cool for about 10 minutes .	Now that you have a hot pan , it 's time to cook the chicken .	<ul style="list-style-type: none"> - The content step 1 is being copied to step 0. - Recovering from incorrect steps until step 2.
	4	For this recipe you will need the following ingredients : - 1 cup of water , salt and pepper	Cut the chicken breast into small pieces .	The first thing you need to do is put the meat in the pan and let it cool for about 10 minutes .	Place the pan on the stove and heat the pan on medium heat .	Make sure it is properly cooked.	Once the chicken is done , remove the chicken from the oven and let it cool for a few minutes	<ul style="list-style-type: none"> - The infilled sentence at step 4 is generic and underspecified
V-Infill	0	You will need the following ingredients	Cut the chicken breast into small pieces .	Add the chicken to the pan .	Heat the oil in a frying pan over medium heat .	Once the chicken is done , add the chicken to the pan and stir it up .	Now that you have a little bit , you will need to put the meat in the oven at 180 degrees celsius circulating air for about 20 minutes .	<ul style="list-style-type: none"> + Step 0 talks about ingredients. - The ingredients are still underspecified.
	4	You will need the following ingredients : - meat , 1 cup of butter and salt	Cut the onion into small pieces .	Cut the chicken breast into small pieces .	Heat the oil in a frying pan over medium heat .	Once the chicken is ready , add the chicken to the pan .	Once the chicken is done , remove the pan from the pan and let it sit for about 10 minutes .	<ul style="list-style-type: none"> + Step 0 talks about ingredients. - Phrase from step 5 copied to step 4

FIGURE 3.16: Comparison of V-Infill and XE dealing with infilling context during inference (for making *chicken roast*). GT corresponds to the ground truth step. The index in each row corresponds to the index of the missing image.

In this section, we present the effects of infilling both during both training and inference on ViST and ViPT datasets. We also present an analysis based on the length of generated sequences along with a qualitative demonstration.

Infilling during training: The overall performance of the models is presented in Table 3.17. Both the infilling model variants achieve higher scores on the recipes while not decreasing their performances on stories. We also observed that increasing the number of masked local features beyond two drastically decreases the performance on both datasets.

Infilling during inference for Visual Procedure Telling: Acquiring parallel pairwise image and narrative data in the wild is often not feasible. Hence, we perform infilling not only at train time but also at inference time to evaluate the ability of the model to bridge contexts when the corresponding image is absent and deal with real-world data imputation scenarios. Table 3.18 demonstrate the performance of the V-Infill model in comparison with the XE model when different indices are infilled during the inference stage. As observed, the

automatic scores get affected detrimentally when the infilled index is to the left, i.e., a lower index. This is because usually the beginning of the sentence comprises introducing the dish followed by listing down the ingredients. For this reason, the density of the number of entities present at the beginning of the procedure is usually higher. Hence reconstructing that from the rest of the recipe is difficult. However, as we move from left to right, i.e., as we gradually increase the infilled index, we observe an increasing trend in the automatic metric.

Infilling during inference for Visual Story Telling: Table 3.19 demonstrates the effects of infilling various indices during inference. This table is analogous to Table 3.18 for stories. As we can see, a similar trend in the increase in all the automatic metrics is present as we move the infill index to the story’s right. While that is still the case, a very interesting observation is that the difference between the performance of XE and Infill models for any given index is much higher for recipes compared to stories. The infilling technique brings much more value to the task when the nature of the text is procedural and dependent more on the surrounding contexts.

Lengths of generated sequences : We compare infilling during inference between baseline XE model and our V-Infill model in Table 3.18. While the METEOR scores remain comparable, the BLEU scores steadily increase as we move the in_{idx} to the right. Specifically, these jumps are bigger after step 3. Quantitatively, this is the result of the model being able to produce longer sequences as we move to the right as BLEU gets penalized for short sentences. Qualitatively, this implies that the initial steps like specifying the ingredients are more crucial than later ones. A similar observation emerges by analyzing the effects of infilling during training. The average length of generated recipes by XE is 71.26 and by V-Infill is 76.49. A similar trend is observed for stories in Table 3.19.

Qualitative Discussion: Figure 3.16 demonstrates an example of generated samples by infilling different indices. The top row shows the steps in the ground truth steps for the corresponding images. The indices on the top row are the indices of the images or the steps, and the indices on the left column (in blue) are the indices whose local features are masked. As observed, the XE model depicts two strategies to recover the missing context. The first is copying the contents that are similar from the adjacent step directly. For instance, while the 0th index of the image is masked, the XE model generates *cutting* from *trimming* and *chicken* from *meat* from the following step. This has nothing to do with the actual description of the corresponding step. However, our V-Infill model is able to generate the sentence depicting that it is listing ingredients in this case. Since the baseline incorrectly generates the first step, it is harder to recover and generate the correct sequence for the rest of the procedure. The second is the strategy of generating generic sentences. When the infilled index is at 4, the baseline model generates a sentence that is generic and not specific to the given set of images. In this case, it generates a statement that says to make sure that it is properly cooked. Our V-Infill model is able to bridge the context from step 3 about heating the oil and step 5 about removing the pan and hence interpolates the missing context to be placing the chicken on the pan.

Despite the recovering strategies used in both these methods, a common problem is observed in the generated steps. The details in the steps are omitted, thereby leading to the problem of *under-specification*. For instance, the actions in step 4 are under-specified by XE when the infilled index is 4. Similarly, the V-Infill model under-specifies the ingredients.

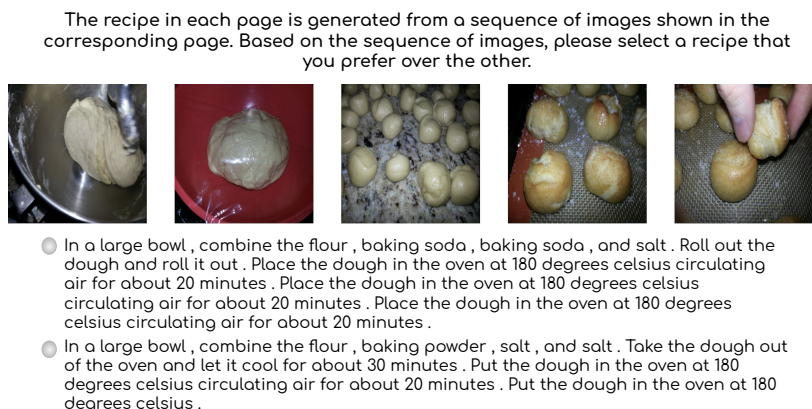


FIGURE 3.17: Human Evaluation Interface for an example of generated recipes with both techniques.

Human Evaluation: Figure 3.17 depicts a screenshot of our human evaluation interface. A sequence of images is presented on top of the screen. This evaluation is conducted to compare between XE and V-Infill models. The generated sentences from both the models, in this case, XE and V-Infill are presented after the images. Note that the generated outputs are presented in arbitrarily random order for each example to ensure there is no bias while performing preference testing. Human subjects are asked to pick one of the generated recipes for the given sequence of images based on the relevance to them. 10 such recipes are presented for each user, and we averaged the preference scores among 20 evaluators.

So far, we explored the N-Local anchoring of content with visual modality. In the next section, we are going to look into the N-Global anchoring of the content driven in the form of a theme provided in the question for biomedical text summarization.

3.5 N-Global Anchoring from Questions

In this section, we describe anchoring a narrative on the whole by providing a topic and specific aspects of the topic in an unstructured format for guidance. As a case study, in this section, we will look at our participation in Phase B of task 5b of the fifth edition of the annual BioASQ challenge, which includes answering factoid, list, yes-no, and summary questions from biomedical data. We describe our techniques with an emphasis on ideal answer generation, where the goal is to produce a relevant, precise, non-redundant, query-oriented summary from multiple relevant documents. The dataset comprises a huge corpus of unstructured text articles, pairs of questions, and gold summaries. In this case, the partial anchor A_i is derived from a question, and the supporting input I_i is the gigantic text corpus. The output narrative N_i is the summary or the ideal answer for the given question from the corpus. The focus of this section is utilizing the anchoring from question to stitch together the appropriate information in the

summary. We use extractive summarization techniques to address this task and experiment with different biomedical ontologies and various algorithms, including agglomerative clustering, Maximum Marginal Relevance (MMR), and sentence compression. We propose a novel word embedding based tf-idf similarity metric and a soft positional constraint that improve our system performance. We evaluate our techniques on test batch 4 from the fourth edition of the challenge. Our best system achieves a ROUGE-2 score of 0.6534 and ROUGE-SU4 score of 0.6536.

In recent years, there has been a huge surge in the number of biomedical articles being deposited online. The National Library of Medicine (NLM) provides MEDLINE, a gigantic database of 23 million references to biomedical journal papers. Approximately 200,000 articles⁵ from this database have been cited since 2015. The rapid growth of information in this centralized repository makes it difficult for medical researchers to manually find an *exact answer* for a question. This section describes our efforts in creating a system that can provide ideal answers for biomedical questions. More specifically, we develop a system that can answer the kinds of biomedical questions present in the dataset for the BioASQ challenge (Tsatsaronis et al., 2015), which is a challenge on large-scale biomedical semantic indexing and question answering. We develop a system for biomedical summarization using MMR and clustering-based techniques.

We build on standard techniques such as Maximal Marginal Relevance (Carbonell and Goldstein, 1998) and Sentence Compression (Filippova et al., 2015) and incorporate domain-specific knowledge using biomedical ontologies such as the UMLS metathesaurus and SNOMEDCT (Stearns et al., 2001) to build an ideal answer generator for biomedical questions. We also experiment with several similarity metrics such as jaccard similarity and a novel word embedding based tf-idf (w2v tf-idf) similarity metric within our system. We evaluate the performance of our system on the dataset for test batch 4 of the fourth edition of the challenge and report our system performance on ROUGE-2 and ROUGE-SU4 (Lin and Hovy, 2003), which are the standard metrics used for official evaluation in the BioASQ challenge. Our best system achieves ROUGE-2 and ROUGE-SU4 scores of 0.6534 and 0.6536 respectively on test batch 4 for task 4b when evaluated on *BioASQ Oracle*⁶. Various configurations and similarity metrics, granularity and algorithms selection enabled us to secure top 1,2,3 in test batch 4 and top 1,2,3,4 in test batch 5 on automatic evaluation metrics of ROUGE-2 and ROUGE-SU4, from our participation in Task 5b of ideal answer generation.

3.5.1 Dataset Description

The training data for Phase B of task 5b provides biomedical questions where each question is associated with question type, URLs of relevant PubMed articles, and relevant snippets from those articles. This dataset consists of 1,799 questions. The A_i anchor for each narrative N_i is derived from the corresponding question Q_i . The narrative N_i which is the summary is then conditioned on this anchor A_i . Though our ideal answer generation system is unsupervised, we use a brief manual inspection of the training data for this edition of the challenge to make an informed choice of hyperparameters for the algorithms used by our system.

⁵https://www.nlm.nih.gov/bsd/medline_lang_distr.html

⁶<http://participants-area.bioasq.org/oracle/>

To develop an ideal answer generator that can produce query-oriented summaries for each question, two popular approaches can be adopted: extractive or abstractive. Extractive summarization techniques choose sentences from relevant documents and combine them to form a summary. Abstractive summarization methods use relevant documents to create a semantic representation of the knowledge from these documents and then generate a summary using reasoning and natural language generation techniques. Brief analysis on a randomly sampled subset from the training data shows us that most of the sentences in the gold ideal answers are present either in the relevant snippets or relevant abstracts of PubMed articles. That is the reason behind adopting extractive summarization for this task. An interesting ordering trend observed among relevant snippets which is used to develop a positional constraint. Adding this positional constraint to our similarity metrics gives us a slight boost in performance. The intuition behind this idea is explained in more detail in §3.5.2.2.

The dataset from test batch 4 of the fourth edition of the BioASQ challenge consisting of 100 questions is used for evaluation.

3.5.2 Model Description

3.5.2.1 N-Global Anchor Representation

The anchors are N-Global in this case, which means they provide a high-level general theme or topic for the narrative. There is no strict explicit fine level governance on every sentence in the summary. The question provides a high-level theme or topic along with the aspects of the entities that are probed about. The biomedical entities in the question still play a major role in determining the topic of the query-based summary. Keeping this in view, we come up with three formulations for the representation of the anchors:

- Surface forms: Words occurring in raw form in the dataset.
- Expansion forms: Biomedical ontologies are leveraged to expand the terms present to the related words.
- Embedding forms: A latent space representation of these words help in designing better scoring functions to evaluate conceptual closeness.

As we can recall, these forms are parallels to the different forms of anchors derived for the complete anchoring task described in §3.3. The corresponding forms for entity skeletons there are *surface forms*, *nominalized forms*, *abstract forms*. Next, we will be seeing how the aforementioned forms for partial anchoring are used in the relevance ranking, which is the first step for content selection. In particular, these different forms influence the way similarity scores are calculated.

3.5.2.2 Summarization Pipeline

In this subsection, we describe our system pipeline for the ideal answer generation task which mainly comprises of three stages: *question-sentence relevance ranker*, *sentence selection* and

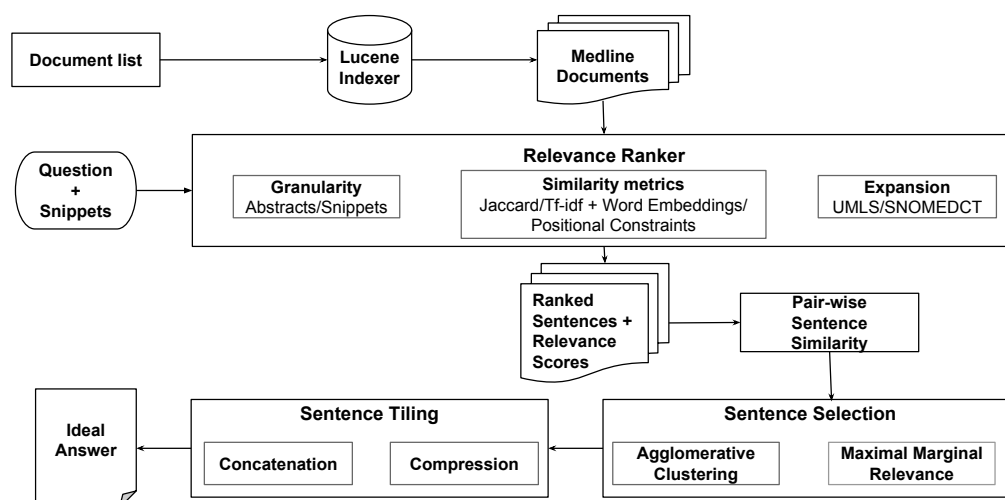


FIGURE 3.18: System pipeline for Ideal Answer Generation (with configuration choices)

sentence tiling. Each stage has multiple configurations depending upon various choices for algorithms, concept expansion, and similarity metrics. The different forms of N-Global anchors play a direct role in concept expansion and the formulation of similarity metrics. Figure 3.18 shows the overall architecture of our system and also briefly mentions various algorithms used in each stage. We describe these stages and choices in more detail in the following subsections. Note that the anchors are all the words in the questions in three forms described in §3.5.2.1. Anchor conditioned summaries are extracted from the relevance ranker and sentence selection.

Question-Sentence Relevance ranker: In this phase, we retrieve a list of candidate sentences from gold abstracts and snippets provided for each question and compute relevance scores with respect to the question for these sentences. We can choose from several similarity metrics, biomedical ontologies, and different granularities for sentence scoring in this stage.

Granularity for Candidate Sentence Extraction The training data provided for the BioASQ task contains a list of PubMed IDs of gold relevant documents from NLM, along with pertinent gold snippets from these documents, for each question. Since the training data only contains PubMed IDs of relevant documents, we extract complete abstract text for these documents by first indexing all Medline abstracts⁷ via Lucene and then retrieving relevant documents based on PubMed IDs.

We now have two choices of granularity for candidate sentence extraction: using entire abstract texts from relevant documents or using only relevant snippets. We experiment with both possibilities. However, since relevant snippets for each question are a subset of abstract texts, which are highly relevant to the question, leveraging this insight and using only snippets for candidate sentence extraction gives us better performance, as we see from the results in §3.5.3.

⁷https://www.nlm.nih.gov/databases/download/pubmed_medline.html

Similarity metrics The performance of both the relevance ranker and the sentence selection phase (which is the following phase in the pipeline) depends on the similarity metrics used to capture question-sentence relevance and sentence-sentence similarity. In this subsection, we describe various similarity metrics which we experiment with. The representation of anchors in various forms affects the way these similarity scores are evaluated.

1. Jaccard similarity: This form of similarity uses the surface form and/or expansion form representation of the N-Global anchor. For each sentence, its relevance with respect to the question is computed as the Jaccard index between the sets containing all words occurring in the question and the sentence. This is the simplest metric that captures surface (word-level) similarity between the question and the sentence. Including related concepts obtained by concept expansion in these word sets provide some measure of semantic overlap, but this technique is not very effective as we show in §3.5.3. Hence this similarity is capable of leveraging the surface form and expansion form representations of the anchors.

2. Tf-idf based similarity with word embeddings: This form of similarity is capable of utilizing the embedding form representations for the N-Global anchor. Using ontologies such as WordNet (for general English) and UMLS/ SNOMEDCT (for biomedical domain) for concept expansion to incorporate some semantics while computing sentence similarity is not sufficient due to the unbounded nature of such ontologies. Hence, to assimilate semantic information in a more controlled manner, we use a novel similarity metric inspired by the widely-used tf-idf cosine similarity metric, which incorporates semantic information by making use of word embeddings (Mikolov et al., 2013).

Let \mathbf{W} represent the symmetric word-to-word similarity matrix and \vec{a}, \vec{b} represent tf-idf vectors for the sentences. The similarity metric is defined as:

$$\text{sim}(\vec{a}, \vec{b}) = \frac{\vec{a}^T \mathbf{W} \vec{b}}{\sqrt{\vec{a}^T \mathbf{W} \vec{a}} \sqrt{\vec{b}^T \mathbf{W} \vec{b}}} \quad (3.4)$$

The word-to-word similarity matrix \mathbf{W} is computed using cosine similarity between word embeddings for each word. We use word embeddings that have been pre-trained on PubMed, PMC, and Wikipedia articles to incorporate domain knowledge⁸.

This form utilizes the embedding form representation of the anchor words.

3. Similarity function with positional constraints: As described in §3.5.1, the data provided for each question contains a list of relevant abstracts of PubMed articles, as well as a list of relevant snippets extracted from these abstracts. The abstracts are ordered by relevance. Snippets, on the other hand, are not ordered by relevance but are ordered according to the abstracts that they are extracted from. Since the abstracts themselves are ordered by relevance, this gives an inherent discourse structure to the snippets. This observation motivates us to incorporate information about a snippet's position in the list into the similarity function to

⁸These pre-trained word vectors are provided by <http://evexdb.org/pmresources/vec-space-models/>

improve the summaries generated by our system. We first test this hypothesis using a simple baseline which gives the first snippet in the list as the summary for every question. This simple baseline is able to achieve good ROUGE scores, as shown in Table 3.21. We experiment with two different ways of incorporating this constraint:

- **Hard positional constraint:** In this method, we enforce the snippet position as a hard constraint. We achieve this by restricting the algorithm to select the first sentence of the summary from the first snippet (most relevant snippet) in the list. The remaining sentences can be selected from any snippet. This method does not have much improvement on our ROUGE scores, as explained in §3.5.3.
- **Soft positional constraint:** This method incorporates the snippet position as a soft constraint by adding it to the similarity function. The augmented similarity function after incorporating the snippet position is presented below:

$$\begin{aligned} \text{positionalSim}(q, s) = \alpha * \text{sim}(q, s) + \\ (1 - \alpha) * \text{rank}(s) \end{aligned} \quad (3.5)$$

Here, q and s denote the question and sentence respectively; $\text{sim}(q, s)$ denotes a function which computes similarity between question and sentence (we experiment with Jaccard and tf-idf based similarities); $\text{rank}(s)$ denotes the boost given to the sentence based on the position of the snippet to which it belongs and α is a weighting parameter. The value of $\text{rank}(s)$ for a sentence is computed as follows:

$$\begin{aligned} \text{rank}(s) &= 1 - \text{pos}(s) \\ \text{pos}(s) &= \text{snippetPos}(s) / \#\text{snippets} \end{aligned}$$

Here, $\text{snippetPos}(s)$ denotes the position (index) of the snippet, to which the sentence belongs, in the list of relevant snippets. If a sentence belongs to multiple snippets, we consider the lowest index. $\#\text{snippets}$ denotes the number of relevant snippets for the current question. This positional boost gives higher weight to sentences with lower position values (since they occur earlier in the list) and returns a normalized value in the range 0-1, to ensure that it is comparable to the range of values produced by the similarity function. Adding this constraint boosts our ROUGE scores.

As we can observe, the constraints are imposed on top of the underlying similarity scores to rerank the retrieved sentences. Hence all the anchor forms compatible in calculating the corresponding $\text{sim}(q, s)$ score can be used along with these positional constraints.

Table 3.20 demonstrates the compatibility of different anchor form representations with the similarity metric computation.

Similarity Metrics	Anchor forms		
	Surface form	Expansion form	Embedding form
Jaccard	✓	✓	
Tf-idf			✓
Positional	✓	✓	✓

TABLE 3.20: Compatibility of anchor form representation with similarity metrics

Biomedical Tools and Ontologies We experiment with various biomedical tools and ontologies for concept expansion in order to incorporate relations between concepts while computing similarity. To perform concept expansion, the first step is to identify biomedical concepts from a sentence. We choose the MetaMap concept identification tool and use a python wrapper, pymetamap⁹ for this purpose. This API identifies biomedical concepts from a sentence and returns a Concept Unique Identification (CUI) for each concept. This CUI acts as a unique identifier for the concept which is shared across ontologies, i.e., it can be used as an ID to retrieve the same concept from the UMLS ontology. After identifying biomedical concepts, we experiment with two ontologies for concept expansion: UMLS Metathesaurus and SNOMEDCT.

- **UMLS Metathesaurus:** The UMLS Metathesaurus contains many types of relations for each biomedical concept. For our task, three relation types are of interest to us: ‘RB’ (broader relationship), ‘RL’ (similar or alike relationship) and ‘RQ’ (related and possibly synonymous relationship). However, none of the biomedical concepts identified from questions and sentences in our training dataset contained relations of the type ‘RL’ or ‘RQ’. Hence we perform expansion for each biomedical concept by collecting all concepts linked to it by the ‘RB’ relation.
- **SNOMEDCT:** The SNOMEDCT ontology does not contain CUIs for biomedical concepts. Hence, we need to use a different technique to locate concepts in this ontology. In addition to CUI, pymetamap also provides a “preferred name” for each concept. We use this preferred name to perform a full-text search in the SNOMEDCT ontology. All concepts returned by this search are then considered to be related concepts and used for expansion. Using this ontology for concept expansion returns a much larger number of related concepts due to the nature of our search (using fuzzy text search instead of precise identifiers).

We use these techniques to perform concept expansion on both questions and sentences from relevant snippets. In §3.5.3.2, we present the results of various system configurations with and without domain-specific concept expansion.

Sentence Selection In this stage, we want to select sentences for the final summary from candidate sentences extracted by the previous stage. Since the BioASQ task has a word limit of 200, we limit the number of sentences selected for the final summary to five. This sentence limit gives us good ROUGE scores across multiple system configurations.

⁹<https://github.com/AnthonyMRios/pymetamap>

The simplest way of performing sentence selection is to continue selecting the sentence with the highest relevance score to the question until the sentence limit is reached. However, sentences with high relevance to the question may be semantically similar, thus introducing redundancy in the generated summary. We use two algorithms to combat this issue: agglomerative clustering based on sentence similarity and Maximum Marginal Relevance (MMR) (Carbonell and Goldstein, 1998). Both algorithms require effective similarity metrics to compute semantic similarity between sentences. We experiment with various similarity metrics described in §3.5.2.2. We also experiment with concept expansion using multiple biomedical ontologies. Selecting relevant content here is anchored in the question.

Agglomerative Clustering Redundancy reduction via clustering is one of the techniques that was proposed for biomedical query-oriented summarization (Chen and Verma, 2006). In this technique, we create all possible sentence pairs from our set of candidate sentences and compute pair-wise similarities. We then perform agglomerative clustering on the sentences using these pair-wise similarity scores. Finally, we select one sentence from each cluster to generate the final summary in such a way that the sentence having maximum question relevance score is selected from every cluster. The number of clusters is set to the maximum number of sentences we need in the final summary (five in this case). The intuition behind this technique is that agglomerative clustering forces semantically similar sentences to fall into the same cluster. Since we only select one sentence from each cluster in the end, we discard sentences that are highly similar to the selected ones.

Maximal Marginal Relevance Maximal Marginal Relevance (Carbonell and Goldstein, 1998) is a widely-used summarization algorithm that was proposed to tackle the issue of redundancy while maintaining query relevance in summarization. This algorithm selects new sentences based on a combination of relevance score with respect to the question as well as similarity score with respect to the sentences which have already been selected for the final summary. Thus, this algorithm incorporates sentence similarity as a constraint instead of explicitly clustering sentences.

Sentence Tiling In the final stage, we combine all selected sentences to produce the final summary. The simplest way is to append all selected sentences while constraining summary length (because of the word-limit constraint for this task). We also experiment with an LSTM-based sentence compression method. We train a neural network based on work done previously by Filippova et al. (2015) for sentence compression. We generate training data for this network by pairing sentences from abstract texts with their full-text versions. Given that this dataset is too small to train the neural network, we add training instances from existing sentence compression datasets. Input to this model includes the word vector representation for a word and a binary value to indicate whether the previous word was included in the output sentence. Based on these inputs, the output of the model predicts whether the word should be deleted or not. In the end, sentences generated after word deletion are concatenated together to generate the final summary. It is to be noted that this model does not require any linguistic features. Note that concatenating the selected content based in the order of relevance does not guarantee a coherent structure in the narrative. This chapter mainly focuses on anchoring

	Experiment	ROUGE-2	ROUGE-SU4
1	Clustering + Abstract texts (with average constraint)	0.2906	0.3138
2	Clustering + Snippets (with average constraint)	0.4314	0.4347
3	Clustering + Snippets (without average constraint)	0.5609	0.5632
4	Clustering + UMLS expansion	0.5488	0.5521
5	Clustering + SNOMEDCT expansion	0.5514	0.5586
6	Clustering + UMLS expansion + weighting	0.5402	0.5431
7	Clustering + SNOMEDCT expansion + weighting	0.5530	0.5588
8	Clustering + UMLS expansion + weighted normalization	0.5592	0.5632
9	Clustering + SNOMEDCT expansion + weighted normalization	0.5585	0.5650
10	MMR	0.6338	0.6296
11	MMR + w2v tf-idf similarity	0.6168	0.6126
12	First snippet baseline	0.3363	0.3308
13	MMR + Hard positional constraint + Jaccard similarity	0.6338	0.6296
14	MMR + Soft positional constraint + Jaccard similarity	0.6419	0.6410
15	Hard positional constraint + Jaccard similarity	0.6328	0.6254
16	Soft positional constraint + Jaccard similarity	0.6433	0.6429
17	Soft positional constraint + w2v tf-idf similarity	0.6534	0.6536
18	MMR + tf-idf similarity + LSTM compression	0.5689	0.5723

TABLE 3.21: ROUGE scores with different algorithms, ontologies and similarity metrics

the content selection in the question. In the next chapter, we will discuss some post-editing techniques to reorder the selected content to form a more coherent summary.

3.5.3 Experiments and Results

We experiment with ideal answer generation using various system configurations which differ in similarity metrics, biomedical ontologies, sentence selection algorithms (clustering/MMR), and tiling algorithms used. The official evaluation for *ideal answers* includes manual evaluation by biomedical experts in the BioASQ team as well as automatic evaluation via ROUGE scores. To present comparable and standardized results, we run our system on the batch 4 dataset for Phase B of task 4b and get our results evaluated via the *BioASQ Oracle*.

3.5.3.1 Quantitative Analysis

These results are shown in Table 1. We obtain the best results among these configurations by using soft positional constraint with tf-idf based similarity on snippets. The first three rows in Table 1 show our experiments with different granularities for sentence extraction. While using abstract texts for sentence selection, we observe that our clustering technique frequently puts sentences with low query relevance into the same clusters. Since our selection method picks one sentence from each cluster, some sentences with low query relevance from these “bad” clusters are also selected for the final summary. To solve this issue, we imposed a constraint that filtered out sentences with a lower-than-average relevance score regarding the question before clustering. We also tried adding this constraint while using relevant snippets, but this reduced our scores because sentences from snippets are already relevant to the question, and

Category	Question/Summary
Question	What is the effect that EZH2 has on chromatin?
Gold Ideal Answer	Ez that catalyzes di- and trimethylation of histone H3 lysine 27 (H3K37me2/3), marks repressive to transcription. [1] The mammalian homologs Ezh1 and Ezh2 form similar PRC2 complexes but exhibit contrasting repressive roles. [2] PRC2-Ezh2 catalyzes H3K27me2/3 and its knockdown affects global H3K27me2/3 levels. [3] EZH2 thus maintains chromatin in a repressive state.
Clustering + Snippets	During progenitor cell differentiation and ageing, PcG silencer EZH2 attenuates, causing loss of PRC binding and transcriptional activation of INK4b and INK4a. Treg-specific ablation of Ezh2 resulted in spontaneous autoimmunity with reduced Foxp3(+) cells in non-lymphoid tissues and impaired resolution of experimental autoimmune encephalomyelitis. We report that the mammalian homologs Ezh1 and Ezh2 form similar PRC2 complexes but exhibit contrasting repressive roles. [2] In this study, we found the inverse correlation between FOXP3 and Ezh2, an enzyme for histone H3K27 trimethylation (H3K27me3) and a central epigenetic regulator in cancer the PRC2-Ezh2 complex, which is bound to the myogenin (MyoG) promoter and muscle creatine kinase (mCK) enhancer in proliferating myoblasts, and the PRC2-Ezh1 complex, which replaces PRC2-Ezh2 on MyoG promoter in post-mitotic myotubes.
Clustering + UMLS expansion + weighted normalization	During progenitor cell differentiation and ageing, PcG silencer EZH2 attenuates, causing loss of PRC binding and transcriptional activation of INK4b and INK4a. We report that the mammalian homologs Ezh1 and Ezh2 form similar PRC2 complexes but exhibit contrasting repressive roles. [2] Treg-specific ablation of Ezh2 resulted in spontaneous autoimmunity with reduced Foxp3(+) cells in non-lymphoid tissues and impaired resolution of experimental autoimmune encephalomyelitis. Ez that catalyzes di- and trimethylation of histone H3 lysine 27 (H3K37me2/3) [3], marks repressive to transcription. [1] the PRC2-Ezh2 complex, which is bound to the myogenin (MyoG) promoter and muscle creatine kinase (mCK) enhancer in proliferating myoblasts, and the PRC2-Ezh1 complex, which replaces PRC2-Ezh2 on MyoG promoter in post-mitotic myotubes.
Clustering + SNOMEDCT expansion + weighted normalization	During progenitor cell differentiation and ageing, PcG silencer EZH2 attenuates, causing loss of PRC binding and transcriptional activation of INK4b and INK4a. Treg-specific ablation of Ezh2 resulted in spontaneous autoimmunity with reduced Foxp3(+) cells in non-lymphoid tissues and impaired resolution of experimental autoimmune encephalomyelitis. We report that the mammalian homologs Ezh1 and Ezh2 form similar PRC2 complexes but exhibit contrasting repressive roles. [2] Ez that catalyzes di- and trimethylation of histone H3 lysine 27 (H3K37me2/3), marks repressive to transcription. [1] the PRC2-Ezh2 complex, which is bound to the myogenin (MyoG) promoter and muscle creatine kinase (mCK) enhancer in proliferating myoblasts, and the PRC2-Ezh1 complex, which replaces PRC2-Ezh2 on MyoG promoter in post-mitotic myotubes.
MMR	Ezh1 and Ezh2 maintain repressive chromatin through different mechanisms. [1] The chromatin-modifying enzyme Ezh2 is critical for the maintenance of regulatory T cell identity after activation. Treg-specific ablation of Ezh2 resulted in spontaneous autoimmunity with reduced Foxp3(+) cells in non-lymphoid tissues and impaired resolution of experimental autoimmune encephalomyelitis. the PRC2-Ezh2 complex, which is bound to the myogenin (MyoG) promoter and muscle creatine kinase (mCK) enhancer in proliferating myoblasts, and the PRC2-Ezh1 complex, which replaces PRC2-Ezh2 on MyoG promoter in post-mitotic myotubes. In this study, we found the inverse correlation between FOXP3 and Ezh2, an enzyme for histone H3K27 trimethylation (H3K27me3) and a central epigenetic regulator in cancer.
MMR + w2v tf-idf	Ezh1 and Ezh2 maintain repressive chromatin through different mechanisms. [1] In this study, we found the inverse correlation between FOXP3 and Ezh2, an enzyme for histone H3K27 trimethylation (H3K27me3) and a central epigenetic regulator in cancer. These studies reveal a critical role for Ezh2 in the maintenance of Treg cell identity during cellular activation. We report that the mammalian homologs Ezh1 and Ezh2 form similar PRC2 complexes but exhibit contrasting repressive roles. [2] The chromatin-modifying enzyme Ezh2 is critical for the maintenance of regulatory T cell identity after activation.
Soft constraint + w2v tf-idf	Ezh1 and Ezh2 maintain repressive chromatin through different mechanisms. [1] We report that the mammalian homologs Ezh1 and Ezh2 form similar PRC2 complexes but exhibit contrasting repressive roles. [2] Ez that catalyzes di- and trimethylation of histone H3 lysine 27 (H3K37me2/3), marks repressive to transcription. During progenitor cell differentiation and ageing, PcG silencer EZH2 attenuates, causing loss of PRC binding and transcriptional activation of INK4b and INK4a. the PRC2-Ezh2 complex, which is bound to the myogenin (MyoG) promoter and muscle creatine kinase (mCK) enhancer in proliferating myoblasts, and the PRC2-Ezh1 complex, which replaces PRC2-Ezh2 on MyoG promoter in post-mitotic myotubes.
MMR + w2v tf-idf + LSTM sentence compression	and ezh2 maintain repressive chromatin through different mechanisms. [1] this study , found the inverse correlation between foxp3 and ezh2 , an enzyme for histone h3k27 trimethylation (h3k27me3) and a central epigenetic regulator in cancer . prc2-ezh2 complex , which is bound to the myogenin (myog) promoter and muscle creatine kinase (mck) enhancer in proliferating myoblasts , and the prc2-ezh1 complex , which replaces prc2-ezh2 on myog promoter in post-mitotic myotubes .

FIGURE 3.19: Summaries generated with different techniques

we end up discarding important information by filtering. We also note that *switching granularity from abstract texts to relevant snippets significantly boosted the ROUGE scores*. Hence all subsequent experiments (rows 4-18) use snippets for sentence extraction.

Rows 4-9 show our experiments with concept expansion using various biomedical ontologies and weighting techniques. We use the following weighting technique: while calculating similarity, words from the original question and sentences carry a weight of 1, while words obtained added after concept expansion carry a weight of 0.5. *We do not observe significant gains using concept expansion*. The unbounded nature of concept expansion hurts our performance, and so we refrain from using this technique in further experiments. Row 10 shows our experiment using MMR for sentence selection instead of clustering. *MMR provides a significant boost in ROUGE score*. Row 11 shows our experiment with the *w2v tf-idf based similarity metric* instead of Jaccard similarity, which *decreases our ROUGE scores slightly but is still better than previous system configurations*. Row 12 shows the scores of a baseline system that returns the first snippet from the list, which is quite high, *validating our assumption that snippet position is an important factor*. Rows 13-17 show our experiments with different ways of adding positional constraints described in §3.5.2.2. While *using a hard constraint does not show much improvement*, *soft positional constraint gives a slight boost*. Results with and without MMR for this metric are nearly comparable. *Soft constraint gives a huge boost when used with w2v tf-idf based*

similarity. Row 18 shows our experiment *adding LSTM-based compression* on top of MMR with w2v tf-idf based similarity, which *reduces our scores*. Row 17 is the system configuration with the highest ROUGE score on our dataset, which uses soft positional constraint with w2v tf-idf similarity.

3.5.3.2 Qualitative Analysis

Figure 3.19 presents ideal answers generated by some of our system configurations for a randomly selected summary question from Task 4b Phase B data to provide a comparative qualitative error analysis. Each sentence in the ideal gold answer is indexed with a number, as shown in the figure. We perform a relative analysis of the extent of information captured by a selected subset of system configurations from Table 1.

The sentence indexed [1] in the gold ideal answer is present word-for-word in summaries created by two configurations: Clustering + SNOMEDCT expansion + weighted normalization and Soft constraint + w2v tf-idf. Clustering + UMLS expansion + weighted normalization contains a longer version of this sentence. We also observe that this sentence does not contain any of the terms from the original question. Hence, summaries generated by all configurations using only Jaccard similarity (Clustering + Snippets, MMR) do not contain this sentence since there is no surface-level similarity. However, methods incorporating semantic information via word embeddings (w2v tf-idf similarity) or concept expansion (UMLS/ SNOMEDCT) include this sentence in the final summary, which shows that incorporating semantic information is important to bridge the vocabulary gap in some situations.

The sentence indexed [2] in the gold answer is present in summaries generated by most of the configurations as shown but with extra phrases such as ‘We report that’ at the beginning of the sentence. Though the presence of such words does not have a major impact on automatic scores like ROUGE, it influences the manual evaluation, which also judges summary readability. However, the LSTM-based compression method removes these words via deletion. We observe that this sentence contains the concept “Ezh2” which is also present in the question. Hence, some configurations which use surface-level similarity (Clustering+Snippets) also pick this sentence for the final summary. But this sentence is not present in the summary generated by the MMR + snippets configuration. This happens because many sentences selected by the algorithm already contain the concept “Ezh2” and so this sentence is excluded due to its similarity to already selected sentences.

3.6 Conclusions and Prospective Future Directions

In this section, we summarize our findings in anchoring the narratives in N-Local and N-Global anchors. From that point, I would like to point out a few potential research directions to place our eggs in for our next steps in this direction.

Automatic narrative generation has been a dream since the emergence of AI. In several circumstances, this narrative is situated in the context. We examined deriving the anchors from this context to improve the narration. In specific, this chapter focused on improving the narrative

by anchoring in relevant content. Anchoring the narrative in content assists in improving the relevance and naturalness aspects. This anchoring can be done at a finer or a coarser level. We presented the N-Local anchoring with visual storytelling and N-Global anchoring with query-oriented summarization.

Scaling image captioning models practically mandates training on noisy and uncurated data available on the web. Our work presents an approach that denoises learning from such large yet diverse web-scaled data with alt-text annotations by sub-selecting content as intermediate skeletons. We experimentally demonstrate that this approach improves the captions significantly in human evaluations on out-of-domain test data by converting meta and story like captions to more visually informative captions. We also demonstrate the transferability of oversimplified English skeleton words to improve captions in five other languages.

Additionally, the natural-language interpretable skeleton layer allows us better control and perform human-in-the-loop corrections of model predictions. We believe this is a promising direction towards unpaired IC and has a strong potential for semi-automatic interventions to correct or interact with the skeletons to guide the final captions better.

Naturalness to a story comes as a package of not only introducing entities, but also referring to them appropriately as humans do. Our work on using N-Local anchor units is inspired by the intuition that humans form a central mindmap of a story before narrating it. In this work, this mindmap is associated with the entities (such as people, location, etc.) involved in the story. We present our work on introducing entity and reference skeletons in generating a grounded story from visual input. In the first phase, we represent the entity skeletons in three forms: surface, nominalized, and abstract. These are different ways to anchor the characters and their referring expressions in the stories. These forms of representations correspond to different properties of the skeleton words like whether they are nouns or pronouns or the category of the nouns based on an ontology. In the second phase, we present a story generation model that takes in the entity skeletons and the images. A strong baseline model is selected that has high performance with respect to human evaluation scores for the task of plain visual storytelling. We extended the baseline to set up another baseline that is informed of the entity skeletons to perform a fair comparison. We then proposed two models: (1) multitasking with the prediction of the skeleton and (2) glocal hierarchical attention model that attends to the skeleton words at the word level and the sentence level hierarchically. We observe that our *MTG* and *glocal hierarchical attention* models can adhere to the skeleton, thereby producing schema-based stories. Our *MTG* model performs better in terms of automatic metrics like METEOR by around 3 units. However, analysis on the percentage of generation of the noun and pronoun forms of entities reveals that the *glocal hierarchical attention* model generates entities closer to the distribution in the ground truth stories. We also conducted human evaluation that reveals that the *glocal hierarchical attention* model is preferred 82% of the time.

We demonstrate that infilling is a simple yet effective technique and a step towards maximizing the utilization of surrounding contexts in visual narratives. Infilling is the strategy that enables the model to learn surrounding contextual information by masking spans of input while the decoding attempts to generate the entire text. To experimentally support our hypothesis, we collect a new large-scale ViPT dataset of 46k procedures comprising 10 categories. We compare the performance of our model and conclusively show the higher significance of infilling-based

techniques in visual procedures compared to visual stories. In the future, we plan to explore the following two directions: (1) interpolating the contexts between consecutive steps by introducing a new infilled image and (2) addressing the underspecification problem by controlling the content in an infilled image with explicit guidance. These infilling techniques are also instrumental when dealing with data imputation with missing contexts and collaborative authoring in real-world scenarios.

Complementing this scenario guides the narrative with an overall theme or topic not divided to provide one-to-one level finer guidance at the sentence level. In this chapter, we presented a system for query-oriented summary generation anchored in question. Since the topic and the aspects in the question drives the general theme of the summary and do not finely govern every sentence, this is the case of N-Global anchoring. The three forms of anchors that we have explored in this section are: surface forms, expansion forms, embedding forms. We observed that incorporating word embedding based tf-idf similarity along with soft positional constraints outperforms surface-level word similarity with soft positional constraints. This is because the former captures both semantic information of the content as well as relevance to query based on sentence position. However, though the expansion forms are bringing more information to the plate, they still seem to hurt the performance. Investigating a controlled utilization of the expansion forms has the potential to reap benefits in improving the content property of narration.

Based on the learning experiences from anchoring the narratives in varied levels of granularities of content, I believe further exploration in any of the following domains has the potential to show significant improvements. These range from anchoring techniques to evaluation which are described in detail as follows:

1. Partial Anchoring: In this chapter, we have explored anchoring in both these extremes, i.e., guidance at the sentence and narrative levels. One of the prospective directions is anchoring the narrative in partial anchoring. This means that instead of anchoring every sentence, we can provide anchors to a group of sentences and another anchor for the next group. For instance, consider a narrative that is segregated into introduction, body and conclusion. With a slightly coarser-grained anchoring compared to N-Local anchors, each of these segregations can be anchored in a theme or a topic.

2. Fusion of N-Local and N-Global anchoring: Instead of categorizing anchors into both ends of the spectrum with respect to the granularity in guiding the narrative, another approach is fusing these anchors. For instance, consider the case where a story is to be weaved around themed characters. The character arcs are provided for every interacting pair of characters, but this information is provided in the form of a theme and not guidance for every sentence. The story is now anchored N-locally with respect to every participating character. However, each character's development is determined on the whole for the entire story.

3. Anchor Form Representation: As we have observed in the case of entity skeletons as N-Local anchors, nominalized representation of the entity skeletons seems to outperform other models in METEOR score. Similarly, in the case of N-Global anchoring, a combination of

expansion and embedding forms has prospective chances of improved performance. Investigating other forms of anchor representation utilized in the generation models is an interesting area with the potential to improve the narratives significantly. For instance, investigating the incorporation of nominalized N-Local anchors in the glocal attention model may reap the benefits of both models. As a natural addition, applying our methods to other forms of conditions to generate storytelling, such as semantic representations, graphs, and prompts, would be interesting to investigate the generalizability of our approaches.

4. Targeted Anchor based Evaluation: Despite the common usage of metrics such as METEOR for text generation tasks, it often lacks the needful targeted for the specific tasks. In the case of N-Local anchoring, we have extended some analysis leveraging the number and the diversity of the entities generated. Similarly, investigating metrics that evaluate the incorporation of anchors in the generated text provides visibility into the specific goal we are attempting to address. Exploring and setting up task-specific metrics to evaluate intermediate tasks such as anchor form or skeleton representation would be useful in streamlining the steps towards the end goal.

Laying Foundation for Anchoring in Structure: So far, we have looked into anchoring the narrative in relevant content in scenarios with the availability of N-Local and N-Global anchors. An important property of the narrative is to identify and approximate the right content. Building on top of this, there is also a necessity for the content to be organized in a coherent layout to enable readability. This property is attributed to the structure, which we are going to look into in the next chapter. Fundamentally, this structure can be used in our framework of N-Local and N-Global anchors.

- *N-Local Anchoring:* Conditioning finely into the entity skeletons anchors the stories into the content. Determining the structural layout is more explicit in procedural how-to activities in comparison to stories. Drawing parallels to the visual storytelling, we gathered a new dataset for storyboarding recipes which has data comprising of parallel image and step-wise textual description. Determining anchors for structure in the same way as content is challenging due to the lack of tangible representation. Hence, as will be examined in the next chapter, we resort to incorporating state sequence information in the generation process to anchor the narrative in structure.
- *N-Global Anchoring:* As we have seen in N-Global anchoring of content, the content selection relies on the general theme provided by the anchor in the question. However, the layout or the organization of the content was performed naively by tiling the sentences based on position or relevance to the anchor. There is potential scope for improving the coherence of the narrative by structuring the selected content in a logical order. In the next chapter, we will explore reordering the sentences and fusing text to anchor the narrative in structure. The reorganization depends on the sentences selected by content; therefore, this layout is determined at the summary level.

4

Structure in Narratives: N-Local and N-Global Anchoring

The confidence people have in their beliefs is not a measure of the quality of evidence but of the coherence of the story the mind has managed to construct.

Daniel Kahneman

In the previous chapter, we have gathered the building blocks needed to gather and induce content-based anchors (both through N-Local and N-Global anchoring) in a narrative. As we have discussed previously, the next step is to arrange or organize the content in a logical form to ensure coherence. This arrangement or coherence contributes to a well-organized thought or construct that builds the evidence in an organized manner.

Determining a high-level layout in terms of structure for the visual stories dataset dealt with in the last chapter is ambivalent. However, this structure is observed definitively in procedural texts such as ‘how-to’ activities. Hence, we collect a dataset that resembles the visual storytelling dataset in a high-level composition for the domain of cooking recipes, which is hereby addressed as ‘*storyboarding recipes*’ dataset. We will discuss anchoring the generation of these recipes in structure in this chapter. In the case of query-based summaries, we primarily relied on the relevance and positions of the individual sentences in the documents from which they

This chapter is based on the following papers:

- “*Storyboarding of Recipes: Grounded Contextual Generation*” (Chandu et al., 2019a)
- “*Extraction meets abstraction: Ideal answer generation for biomedical questions*” (Li et al., 2018c)

were extracted. This is used in the form of positional constraints, which is incorporated in content selection. However, the problem of assimilating an overall view with respect to the other sentences and relative to the central idea of the rest of the narrative remains. This is addressed in this chapter by accounting the overall view of the selected sentences by reordering and fusing them as a post-editing step. We are going to delve deeper into the techniques used for these N-Local and N-Global anchoring forms. Along similar lines in the previous chapter, our focus is not efficiently deriving these structural anchors. We use existing techniques to define and represent structural layouts of the narratives.

A differentiating factor between anchoring in the content and anchoring in structure is the tangibility of the anchor. The former has tangible units to be incorporated in the generation sequence, thereby leading up to tangible metrics to optimize in the loss. In the case of structure, similar anchors are not tangible units present in the form of skeleton chains.

As we have seen in the previous chapter, N-Local anchoring provides multiple units of anchors to aid and drive the generation. This means that $\mathbf{A}_i = \{\mathbf{a}_i^{(1)}, \mathbf{a}_i^{(2)}, \dots, \mathbf{a}_i^{(k)}\}$. In the case of anchoring in content, the individual $\mathbf{a}_i^{(j)}$, where $j \in [1, k]$, is a tangible or interpretable unit. This tangibility is not trivially observed in the case of structure. So, here \mathbf{A}_i is the fine-grained N-Local sequence of structural anchor units. The innumerable possibilities of organizing content quickly result in a combinatorial explosion of state space. Hence, we use clustering techniques to represent different phases, and then finite state machines to organize these phases into state sequences. This state sequence information is used to guide the generation of every step in the recipes and is hence completely anchored in and guided by structure. Structural anchors in this case are hence represented in two forms: *phase clusters* and *state sequences*, which are the possibilities that \mathbf{A}_i sequence can take. These representations, along with the way they are incorporated in the generation process, are detailed in §4.2 of this chapter.

In contrast, for the case of N-Global anchoring, this guidance is not granularly provided by the input. The guidance can be derived in a pseudo-form at a narrative level. Clustering techniques are very useful for reducing the explosion of state space in N-Local and N-Global anchoring. However, in some cases, the structural layout of the narrative is not dictated by granular input. This is observed in the case of query-oriented summarization. In the case of storyboarding, the governing structure is provided for each step in the how-to procedure. This is not the case for the topic or question provided in the query-based summaries. The structure is dictated indirectly from the question based on the relevant content extracted. This is based on the hypothesis that units belonging to the same or similar sub-content should appear together to ensure a smooth flow between the topics of the content. Hence, we extend the techniques used in anchoring content in the previous chapter to indirectly drive the organization of the content to form a logical narrative. The similarity or redundancy of the content units extracted anchors the reorganization of the relevant material. Hence \mathbf{A}_i is a function of \mathbf{I}_i that dictates the layout of \mathbf{N}_i . The question is not directly providing guidance to every sentence. Rather the selected content units are rearranging among themselves with indirect or partial governance from the question. Our focus in this sub-part is going to be mainly on ordering the sentences in the text gathered from anchoring in content. This is detailed in §5.4.

As we will see in the next section, there are two paradigms to deal with organizing the material to maintain a coherent structure. The first is *no pre-selection of content*. We take this paradigm

to deal with content and structure within the same model with explicit N-Local anchoring for structural representations. The other end of the spectrum is *pre-selection of content*. We approach this paradigm with N-Global anchoring as the whole of the content, or the narrative is selected. The task at hand now is to reorganize the material to make it logical and coherent.

4.1 Related Work

Anchoring based on Domain: Martin et al. (2017) and Khalifa et al. (2017) demonstrated that the predictive ability of a seq2seq model improves as the language corpus is reduced to a specialized domain with specific actions. Our choice of restricting domain to recipes is inspired by this, where the set of events is specialized (such as ‘cut’, ‘mix’, ‘add’) although we do not explicitly use event representations. These specialized set of events are correlated to phases of procedural text as described in the following sections. Specifying the domain of text sometimes can induce a structure and thereby a structural skeleton that aids generation. These specialized actions form an inherent sequence either from domain knowledge or learning the sequence from data.

Planning while writing: Traditionally, planning the content before writing is what draws parallels to the whole idea of generating text from anchors. Simply put, the plan here is equivalent to our anchors. A major challenge faced by neural text generation (Lu et al., 2018a) while generating long sequences is the inability to maintain structure, contravening the coherence of the overall generated text. This aspect was also observed in various tasks like summarization (Liu et al., 2018b), story generation (Fan et al., 2019a). (Yao et al., 2018a) experimented with static and dynamic schema to realize the entire storyline before generating.

Pre-selection of Content: There are two schools of modeling approaches with respect to this. The first does not pre-select the content or in this context ‘data’ on which the generation process relies. By this, I mean that this is not explicitly modeled, although techniques like attention are implicitly performed to give more weight or importance to certain content. One such piece of work that relies on pre-selecting content and then planning accordingly was explored by Puduppully et al. (2018) in data to text generation. This sometimes entails learning similarity or analogies between the formatted data available in organized structures and free-form texts. Perez-Beltrachini and Lapata (2018) have learnt these alignments loosely to bootstrap the training process. They have addressed this problem using multi-instance learning.

No Pre-selection of Content: The other end of the spectrum does not select content and then explicitly build a planning algorithm. This paradigm of algorithms typically follows ‘generate as we go’. For instance, Wong and Mooney (2007) show that a hybrid model that takes in forward and inverted mappings from phrase-based statistical machine translation methods is effective in generating text which does not make use of an explicit planning algorithm. It is worth mentioning another section of work in this context that selects content but omits any

planning. Similarly, [Belz \(2008\)](#) proposed a non-modular and integrated framework to generate weather forecasts from probabilistic space automatically. The two processes of content selection and surface realization are combined by [Konstas and Lapata \(2012\)](#) with probabilistic context-free grammar represented as a weighted hypergraph. The generation process is reframed as finding the best derivation tree from this graph. However, this technique is challenged in dealing with long-form narrative descriptions. However, in this work, we propose a hierarchical multi-task approach to perform structure-aware generation.

Ordering Content in text: [Sha et al. \(2018\)](#) proposed an order planning text generation algorithm to generate biographies from Wikipedia data. The heuristic location-based addressing with external memory by [Graves et al. \(2016\)](#) is modeled with a likelihood of a field given the previous fields. [Agrawal et al. \(2016\)](#) introduced the task of sorting a temporally jumbled set of image-caption pairs from a story such that the output sequence forms a coherent story.

Comprehending Food: Recent times have seen large scale datasets in food, such as Recipe1M ([Marin et al., 2018](#)), Food-101 ([Bossard et al., 2014](#)). Food recognition ([Arora et al., 2019](#)) addresses understanding food from a vision perspective. [Salvador et al. \(2018\)](#) worked on generating cooking instructions by inferring ingredients from an image. [Zhou et al. \(2018e\)](#) proposed a method to generate procedure segments for YouCook2 data. In the NLP domain, this is studied as generating procedural text by including ingredients as checklists ([Kiddon et al., 2016](#)) or treating the recipe as a flow graph ([Mori et al., 2014](#)). Our work is at the intersection of two modalities (language and vision) by generating procedural text for recipes from a sequence of images. [Bosselut et al. \(2017\)](#) worked on reasoning non-mentioned causal effects, thereby improving the understanding and generation of procedural text for cooking recipes. This is done by dynamically tracking entities by modeling actions using state transformers.

4.2 N-Local Anchoring from phases in recipes

Structural layouts of narratives are more apparent in goal-oriented procedural texts such as various ‘how-to’ activities. In this regard, we introduce a dataset for sequential procedural (*how-to*) text generation from images in the cooking domain. The dataset consists of 16,441 cooking recipes with 160,479 photos associated with different steps. We set up a baseline motivated by the best performing model in terms of human evaluation for the Visual Story Telling (ViST) task. In addition, we introduce two models in this section to incorporate high-level structure learnt by a Finite State Machine (FSM) in neural sequential generation process by: (1) Scaffolding Structure in Decoder (SSiD) (2) Scaffolding Structure in Loss (SSiL). Our best-performing model (SSiL) achieves a METEOR score of 0.31, which is an improvement of 0.6 over the baseline model. We also conducted a human evaluation of the generated grounded recipes, which reveal that 61% found that our proposed (SSiL) model is better than the baseline model in terms of overall recipes. We also discuss the analysis of the output, highlighting key important NLP issues for prospective directions.

Interpretation is heavily conditioned on context. Real-world interactions provide this context in multiple modalities. In this section, this context is derived from visual features and language.

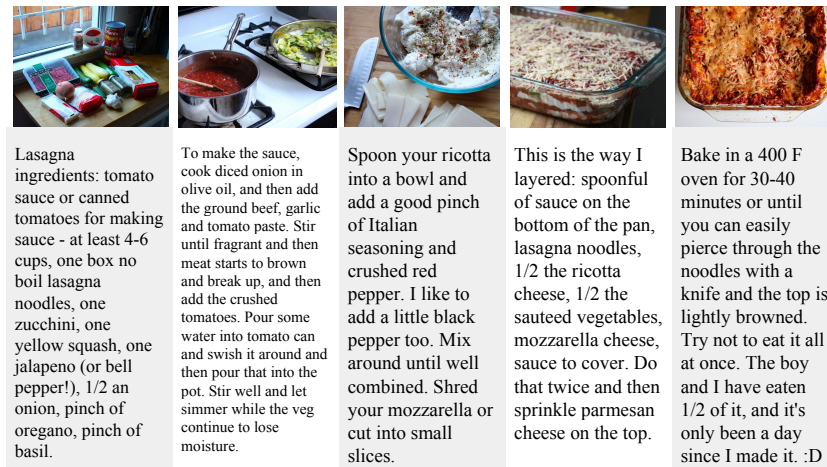


FIGURE 4.1: Storyboard for the recipe of vegetable lasagna

The description of a picture changes drastically when seen in a sequential narrative context. Formally, this task is defined as: given a sequence of images $\mathbb{I} = \{I_1, I_2, \dots, I_n\}$ and pairwise associated textual descriptions, $\mathbb{T} = \{T_1, T_2, \dots, T_n\}$; for a new sequence \mathbb{I}' , our task is to generate the corresponding \mathbb{T}' . Figure 4.1 depicts an example for making *vegetable lasagna*, where the input is the first row and the output is the second row. We call this a ‘*storyboard*’, since it unravels the most important steps of a procedure associated with corresponding natural language text. The sequential context differentiates this task from image captioning in isolation. The dataset is similar to that of ViST (Huang et al., 2016) with an apparent difference between stories and instructional in-domain text, which is the clear transition in phases of the narrative. This task supplements the task of ViST with a richer context of goal-oriented procedure (*how-to*). Numerous online blogs and videos depict various categories of *how-to* guides for games, do-it-yourself (DIY) crafts, technology, etc. This task lays initial foundations for full fledged storyboarding of a given video by selecting the right junctions/clips to ground significant events and generate sequential textual descriptions. We are going to focus on the domain of cooking recipes in the rest of this section. We discuss our approach in generating more structural/coherent cooking recipes by explicitly modeling the state transitions between different stages of cooking (phases). We introduce a framework to apply traditional FSMs to incorporate more structure in neural generation.

The two main contributions of this section are: (1) A dataset of 16k recipes targeted for sequential multimodal procedural text generation, (2) Two models (SSiD: Structural Scaffolding in Decoder ,and SSiL: Structural Scaffolding in Loss) for incorporating high-level structure learnt by an FSM into a neural text generation model to improve structure/coherence. This structure is N-locally induced in the generation model.

4.2.1 Data Collection and Description

We identified two *how-to* blogs from: *instructables.com* and *snapguide.com*, comprising step-wise instructions (images and text) of various *how-to* activities like games, crafts, etc.. We gathered 16,441 samples with 160,479 photos for food, dessert, and recipe topics. We used 80% for training, 10% for validation, and 10% for testing our models. In some cases, there are

Data Sources	# Recipes	# Avg Steps
<i>instructables</i>	9,101	7.14
<i>snapguide</i>	7,340	13.01

TABLE 4.1: Details of dataset for *storyboarding* recipes

multiple images for the same step, and we randomly select an image from the set of images. We indicate that there is a potential space for research here in selecting the most distinguishing/representative/meaningful image. Details of the datasets are presented in Table 4.1. The data and visualization of the distribution of topics are here¹. A trivial extension could be done on other domains like gardening, origami crafts, fixing guitar strings, etc., which is left for future work.

4.2.2 Models Description

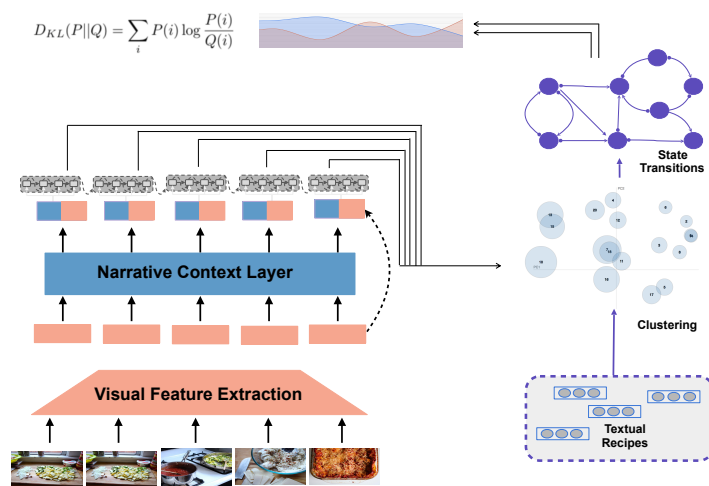


FIGURE 4.2: Architecture for incorporating high level structure in neural recipe generation by n-local anchoring through state machines

We first describe a baseline model for the task of storyboarding cooking recipes in this section. We then propose two models with incremental improvements to incorporate the structure of procedural text in the generated recipes: SSiD (Scaffolding Structure in Decoder) and SSiL (Scaffolding Structure in Loss). The architecture of *scaffolding* structure is presented in Figure 4.2, of which different aspects are described in the following subsections. The anchors for structure are not tangible, thereby challenging the representation of different forms of the anchors. Three different forms of the structural anchors are used in the following scaffolding models, namely: *hard phases*, *hard states* and *soft states*. These are described in detail as and when we model and use them. The indexed representations of these forms corresponding to the models are depicted in §4.2.3.

¹<https://storyboarding.github.io/story-boarding/>

1. Baseline Model (Glocal): The baseline model is inspired from the best performing system in the ViST challenge with respect to human evaluation (Kim et al., 2018). The images are first resized into 224 X 224. Image features for each step are extracted from the penultimate layer of pre-trained ResNet-152 (He et al., 2016b). These features are then passed through an affinity layer to obtain an image feature of dimension 1024. To maintain the context of the entire recipe (global context), the sequence of these image features is passed through a two-layered Bi-LSTM with a hidden size of 1024. To maintain specificity of the current image (local context), the image features for the current step are concatenated using a skip connection to the output of the Bi-LSTM to obtain a glocal representation. Dropout of 0.5 is applied systematically at the affinity layer to obtain the image feature representation and after the Bi-LSTM layer. Batch normalization is applied with a momentum of 0.01. This completes the encoder part of the sequence to sequence architecture. These glocal vectors are used for decoding each step. These features are passed through a fully connected layer to obtain a representation of 1024 dimension, followed by a non-linear transformation using ReLU. These features are then passed through a decoder LSTM for each step in the recipe, which is trained by teacher forcing. The overall coherence in the generation is addressed by feeding the decoder state of the previous step to the next one. This is a seq2seq model translating one modality into another. The model is optimized using Adam with a learning rate of 0.001 and weight decay of 1e-5.

The model described above does not explicitly cater to the structure of the narration of recipes in the generation process. However, we know that procedural text has a high-level structure that carries a skeleton of the narrative. In the subsequent subsections, we present two models that impose this high-level narrative structure as a scaffold. While this scaffold lies external to the baseline model, it functions on imposing the structure in the decoder (SSiD) and the loss term (SSiL).

2. Scaffolding Structure in Decoder (SSiD): There is a high-level latent structure involved in a cooking recipe that adheres to transitions between steps that we define as *phases*. Note that the steps and phases are different here. To be specific, according to our definition, one or more steps map to a phase (this work does not deal with multiple phases being a part of a single step). Phases may be ‘listing ingredients’, ‘baking’, ‘garnishing’ etc., The key idea of the SSiD model is to incorporate the sequence of phases in the decoder to impose structure during text generation.

There are two sources of supervision to drive the model: (1) multimodal dataset $\mathbb{M} = \{\mathbb{I}, \mathbb{T}\}$ from §4.2.1, (2) unimodal textual recipes² \mathbb{U} to learn phase sequences. Finer phases are learnt using clustering followed by an FSM.

Clustering: K-Means clustering is performed on the sentence embeddings with compositional n-gram features (Pagliardini et al., 2018) on each step of the recipe in \mathbb{U} . Aligning with our intuition, when k is 3, it is observed that these clusters roughly indicate categories of *desserts*, *drinks* and *main course foods* (*pizza*, *quesadilla* etc.). However, we need to find out finer categories of the phases corresponding to the phases in the recipes. We use k-means clustering to obtain the categories of these phases. We experimented with different number

²www.ffts.com/recipes.htm

of phases \mathbb{P} as shown in Table 4.2. For example, let an example recipe comprise of 4 steps i.e, a sequence of 4 images. At this point, each recipe can be represented as a hard sequence of phases $\mathbf{r} = \langle p_1, p_2, p_3, p_4 \rangle$.

FSM: The phases learnt through clustering are not ground truth phases. We explore the usage of an FSM to individually model hard and a softer representation of the phase sequences by leveraging the states in an FSM. We first describe how the hard representation is modeled. The algorithm was originally developed for building language models for limited token sets in grapheme to phoneme prediction. The iterative algorithm starts with an ergodic state for all phase types and uses entropy to find the best state split that would maximize the prediction. As opposed to phase sequences, each recipe is now represented as a state sequence (decoded from FSM) i.e, $\mathbf{r} = \langle s_1, s_2, s_3, s_4 \rangle$ (hard states). This is a hard representation of the sequence of states.

We next describe how a soft representation of these states is modeled. Since the phases are learnt in an unsupervised fashion, and the ground truth of the phases is not available, we explored a softer representation of the states. We hypothesize that a soft representation of the states might smooth the irregularities of phases learnt. From the output of the FSM, we obtain the state transition probabilities from each state to every other state. Each state s_i can be represented as $\langle q_{ij} \forall j \in \mathcal{S} \rangle$ (soft states), where q_{ij} is the state transition probability from s_i to s_j and \mathcal{S} is the total number of states. This is the soft representation of state sequences.

The structure in the recipe is learnt as a sequence of phases and/or states (hard or soft). This is the structural *scaffold* that we would like to incorporate in the baseline model. In the SSiD model, for each step in the recipe, we identify which phase it is in using the clustering model and use the phase sequence to decode state transitions from the FSM. The state sequences are concatenated to the decoder in the hard version, and the state transition probabilities are concatenated in the decoder in the soft version at every time step.

At this point, we have 2 dimensions; one is the complexity of the phases (\mathcal{P}), and the other is the complexity of the states in FSM (\mathcal{S}). Comprehensive results of searching this space are presented in Table 4.2. We plan to explore the usage of a hidden markov model in place of FSM in the future.

3. Scaffolding Structure in Loss (SSiL): In addition to imposing structure via SSiD, we explored measuring the deviation of the structure learnt through phase/state sequences from the original structure. This leads to our next model, where the deviation of the structure in the generated output from that of the original structure is reflected in the loss. The decoded steps are passed through the clustering model to get phase sequences, and then state transition probabilities are decoded from FSM for the generated output. Going a step further, we investigate the divergence between the phases of generated and original steps. This can also be viewed as hierarchical multi-task learning (Sanh et al., 2018). The first task is to decode each step in the recipe (which uses a cross entropy criterion, \mathbf{L}_1). The second task uses KL divergence between phase sequences of decoded and original steps to penalize the model (say, \mathbf{L}_2). When there are τ steps in a recipe, we obtain $o(s_1^\tau)$ and $g(s_1^\tau)$ as the distributions of phases

FST Complexity	1	20	40	60	80	100	120
20 Phases	11.27	11.60	12.31	13.71	12.32	12.51	12.36
40 Phases	12.03	12.44	11.48	12.58	12.50	13.91	11.82
60 Phases	11.13	11.18	12.74	12.26	12.47	12.98	11.47

TABLE 4.2: BLEU Scores for different number of phases (\mathbb{P}) and states (\mathbb{S})

comprising of soft states for the original and generated recipes respectively. We measure the KL divergence (D_{KL}) between these distributions:

$$D_{KL}(o(s_1^\tau) || g(s_1^\tau)) = \sum_{i=1}^{\tau} \sum_{j=1}^S o(s_i[j]) \log \frac{o(s_i[j])}{g(s_i[j])}$$

Each task optimizes different functions, and we minimize the combination of the two losses.

$$\sum_{I, T \in \mathbb{I}, \mathbb{T}} \mathbf{L}_1(I, T) + \alpha \sum_{U \in \mathbb{U}} \mathbf{L}_2(U)$$

This combined loss is used to penalize the model. Here, α is obtained from the KL annealing (Bowman et al., 2015) function that gradually increases the weight of the KL term from 0 to 1 during train time.

4.2.3 N-Local Anchor Representation: Phases and States

The anchors for structure are not tangible surface-level words that can be extracted and utilized for conditioned generation. The intangible underlying layout is represented in the forms of phases and states that are discussed in the previous subsections. There are three forms of structural anchor representations distributed among states and phases along with hard or soft representations.

- *Hard phases:* $\langle p_1, p_2, p_3, p_4 \rangle$
- *Hard States:* $\langle s_1, s_2, s_3, s_4 \rangle$
- *Soft States:* $\langle \langle q_{11}, \dots, q_{1S} \rangle, \langle q_{21}, \dots, q_{2S} \rangle, \langle q_{S1}, \dots, q_{SS} \rangle \rangle$, where q_{ij} is the state transition probability from s_i to s_j

The condensed forms to compare between the aforementioned models are presented in Figure 4.3. The SSiL model is a combination of latent modeling and multitasking, as shown here.

4.2.4 Experiments and Results

4.2.4.1 Quantitative Analysis

The two dimensions explored in FSM are \mathbf{P} and \mathbf{S} and exhaustive results are presented in Table 4.2. The BLEU score (Papineni et al., 2002b) is the highest when \mathbf{P} is 40 and \mathbf{S} is 100. Fixing

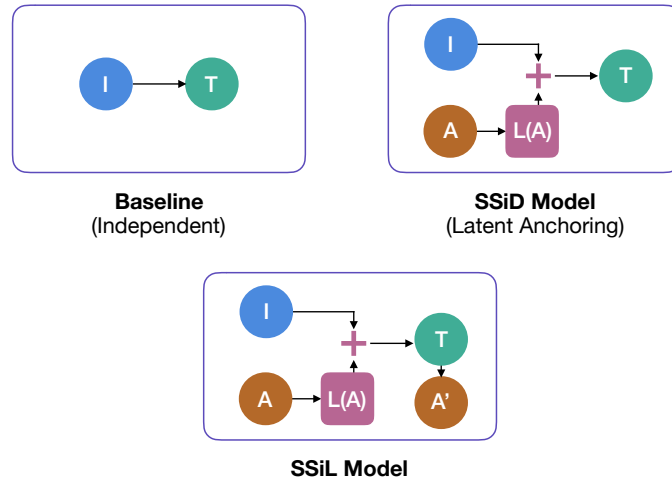


FIGURE 4.3: Comparison of the structure-anchoring based on multimodal interactions

Models	BLEU	METEOR	ROUGE-L
Glocal	10.74	0.25	0.31
SSiD (hard phases)	11.49	0.24	0.31
SSiD (hard states)	11.93	0.25	0.31
SSiD (soft phases)	13.91	0.29	0.32
SSiL (soft phases)	16.38	0.31	0.34

TABLE 4.3: Evaluation of storyboarding recipes

Models					Phenomena
<i>Glocal Model</i>	This is a simple recipe for making a delicious chicken salad.	You will need: a butter knife a plate of bread flour a little bit of salt a dash of pepper flakes a couple of tablespoons of olive oil a pinch of sugar.	Add butter evenly on the pan.	Put the chicken on the grill and set aside.	<ul style="list-style-type: none"> - Ingredients phase wrongly identified. - Wrong ingredients. - Improper conclusion.
<i>SSiD Model</i>	This is a simple recipe for making a delicious and easy dish.	Ingredients: 4 pounds chicken 2 tsp salt, ½ tsp sugar, marinara sauce, mozzarella cheese (i used provolone). Tools: a knife, an oven for the chicken, tongs. Mix all ingredients in a bag.	Add butter evenly on the pan.	Serve the baked chicken wings and enjoy the evening!	<ul style="list-style-type: none"> - Learnt majority structure (step 1) + Got 'tongs' right because of separate tools mention. - The action of baking is not explicitly mentioned (before 'baked' wings).
<i>SSiL Model</i>	You will need: 5 pounds of chicken wings, ½ cup all purpose flour, ½ tsp salt, 2 tsp of paprika, melted butter, silicon mat, baking pan.	Preheat oven to 450 F. Mix dry ingredients in the dry ziplock bag.	Place a mat on the baking pan and spread butter evenly on it.	Spread the chicken pieces on butter on the baking pan. Bake until crispy for 30 minutes. Serve and enjoy!	<ul style="list-style-type: none"> + Global context of baking maintained in preheating. + Non-repetitive ingredients phase. + Referring expressions (baking pan -> it). - Not mentioned tools (tongs).

FIGURE 4.4: Comparison of generated storyboards for *Easy Oven Baked Crispy Chicken Wings*

these values, we compare the models proposed in Table 4.3. The models with hard phases and hard states are not as stable as the one with soft phases since backpropagation affects the impact of the scaffolded phases. Upon manual inspection, a key observation is that for SSiD model, most of the recipes followed a similar structure. It seemed to be conditioned on a global structure learnt from all recipes rather than the current input. However, the SSiL model seems to generate a recipe that is conditioned on the structure of that particular example.

Human Evaluation: We have also performed human evaluation by conducting a user preference study to compare the baseline with our best performing SSiL model. We randomly sampled generated outputs of 20 recipes and asked ten users to answer two preferences: (1)

overall recipe based on images, (2) structurally coherent recipe. Our SSiL model was preferred 61% and 72.5% for the overall and structural preferences respectively. This shows that while there is a viable space to improve the structure, generating an edible recipe needs to be explored to improve the overall preference.

4.2.4.2 Qualitative Analysis

Figure 4.4 presents the generated text from the three models with an analysis described below.

Coherence of Referring Expressions: Introducing referring expressions is a key aspect of coherence (Dale, 2006, 1992), as seen in the case of *'baking pan'* being referred as *'it'* in the SSiL model.

Context Maintenance: Maintaining overall context explicitly affects generating each step. This is seen in the SSiL model where *'preheating'* in the second step is learnt from *'baking'* step that appears later although the image does not show an oven.

Schema for Procedural Text: Explicit modeling of structure has enabled the SSiD and SSiL models to conclude the recipe by generating words like *'serve'* and *'enjoy'*. Lacking this structure, the glocal model talks about *'setting aside'* at the end.

Precision of Entities and Actions: The SSiD model introduces *'sugar'* in ingredients after generating *'salt'*. A brief manual examination revealed that this co-occurrence is a common phenomenon. The SSiL model misses *'tongs'* in the first step.

In this section, we have explored N-Local anchoring of structure in a generate as we go paradigm. In the next section, we are going to look into N-Global anchoring in structure based on the paradigm of pre-selection of content.

4.3 N-Global Anchoring from Reordering

As we have seen in the previous chapter, the growing number of biomedical publications is a challenge for human researchers, who invest considerable effort to search for relevant documents and pinpointed answers. Moreover, extractive summarization techniques, which concatenate the most relevant text units drawn from multiple documents, perform well on automatic evaluation metrics like ROUGE, but score poorly on human readability, due to the presence of redundant text and grammatical errors in the answer. Evaluation based on such metrics for anchoring in content does not imply the overall quality of the narratives. The complementary anchoring that we focus on in this section is in structure. This is an extension to the work we have explored in N-Global anchoring in content in the previous chapter. The relevant content is extracted on the whole from the question. We have performed naive concatenation

based on relevance in the previous chapter. In this chapter we treat this as pre-selected content and reorganize the material at the narrative level. It is implicitly driven by the content extracted based on the question. The question does not have an explicit contribution in determining the structure of the summary, rather has an implicit influence in the form of content derived. In this section, we will discuss three novel approaches for sentence ordering from the sentences selected by anchoring in content. Along with this, our experiments on sentence fusion based on Integer Linear Programming are presented. This is an attempt to improve the human readability of ideal answers.

Human researchers invest considerable effort when searching very large text corpora for answers to their questions. Existing search engines like PubMed (Falagas et al., 2008) only partially address this need since they return relevant documents but do not provide a direct answer for the user’s question. The process of filtering and combining information from relevant documents to obtain an ideal answer is still time consuming (Tsatsaronis et al., 2015). Biomedical Question Answering (BQA) systems can automatically generate ideal answers for a user’s question, significantly reducing the effort required to locate the most relevant information in a large corpus.

Our goal is to build an effective BQA system to generate coherent, query-oriented, non-redundant, human-readable summaries for biomedical questions. Our approach is based on an extractive BQA system (Chandu et al., 2017a) which is used as stitching an ideal answer together by anchoring in content in the previous chapter. However, owing to the extractive nature of this system, it suffers from problems in human readability and coherence. In particular, extractive summaries which concatenate the most relevant text units from multiple documents are often incoherent to the reader, especially when the answer sentences jump back and forth between topics. Although the existing extractive approach explicitly attempts to reduce redundancy at the sentence level (via SoftMMR), stitching together existing sentences always admits the possibility of redundant text at the phrase level. In this section, we will discuss an improvement upon the baseline extractive system in two ways:

- Re-ordering the sentences that are selected by anchoring in content. Since the guidance from content implicitly affects the structure of the summary at the entire narrative level, the question provides N-Global anchoring in structure.
- Fusing words and sentences to form a more human readable summary. This also serves as a post-editing step to smoothen the process the summary put together.

4.3.1 Models Description

The data is the same as described in the previous chapter. So, first, start by looking into an overview of the baseline system.

Overview of Baseline System: Anchored in Content In this section, we provide a brief layout of our baseline system, which achieved the top ROUGE scores in the final test batches

of the fifth edition of BioASQ Challenge (Chandu et al., 2017a). This system includes baseline modules for relevance ranking, sentence selection, and sentence tiling.

The relevance ranker of the baseline performs the following steps: 1) Expand concepts in the original question using a metathesaurus, such as UMLS (Bodenreider, 2004) or SNOMEDCT (Donnelly, 2006); and 2) calculate a relevance score (e.g., Jaccard similarity) for each question/snippet pair (to measure relevance) and each pair of generated snippets (to measure redundancy). The sentence selection model of the baseline used the Maximal Marginal Relevance (MMR) algorithm (Carbonell and Goldstein, 1998), which iteratively selects answer sentences according to their relevance to the question and their similarity to sentences that have already been selected until a certain number of sentences have been selected. The sentence tiling module of the baseline simply concatenates selected sentences up to a given limit on text length (200 words), with no explicit attempt to improve the coherence of the resulting summary.

The baseline system achieved high ROUGE scores but performed poorly on the human readability evaluation in BioASQ 2017. In order to improve human readability, we first developed several post-processing modules, such as sentence re-ordering and sentence fusion, which will be discussed in detail in the following sections.

Modeling Approach: Sentence Ordering

Motivation: We tried to improve upon the Soft MMR system (Chandu et al., 2017a). This pipeline assumes the relevance to be a proxy for ordering the selected sentences to generate the final summary. On the other hand, it does not take into account the flow and transition of sentences to build a coherent flow between these sentences. Since the maximum length of the answer is 200 words (as imposed by the competition guidelines), this system optimizes on selecting the most non-redundant query relevant sentences to maximize the ROUGE score. In this section, we focus on different types of sentence ordering that lead to more coherent answers.

1. Similarity Ordering: The intuition behind the Similarity Ordering algorithm is that sentences that have similar content should appear consecutively so that the generated answer is not jumping back and forth between topics. Our implementation is based on work by Zhang (2011), which discusses the use of similarity metrics at two levels - first to cluster sentences, and then to order them within a cluster - which can lead to big improvements in coherency and readability. We apply this approach to the BQA domain, where we cluster our set of candidate answers using k -means with $k = 2$. We then order the sentences within each cluster, starting with the candidate sentence nearest to the centroid of its cluster and working outward. The intuition is that the most central sentence will contain the largest number of tokens shared by all the sentences in the cluster and is likely to be the most general or comprehensive sentence. This supports our goal of an ideal answer that begins with a broad answer to the question, followed by specifics and supporting evidence from the literature.

In Figure 4.5a, we see that the order of the sentences that appear in the final answer is completely independent of their ordering in the original snippets.

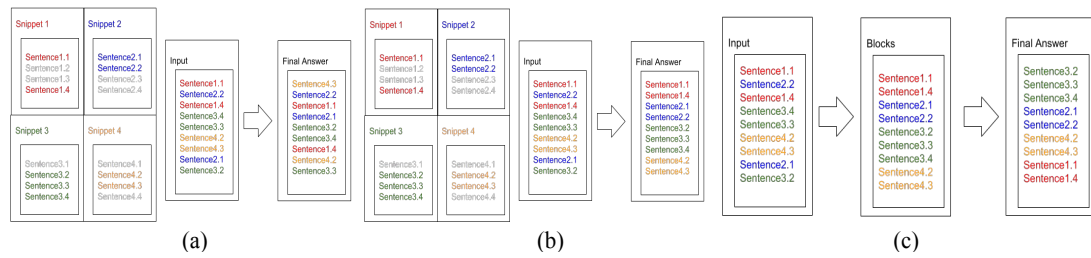


FIGURE 4.5: (a) Similarity Ordering (b) Majority Ordering (c) Block Ordering

2. Majority Ordering: The Majority Ordering algorithm (Barzilay and Elhadad, 2002) makes two main assumptions that are quite reasonable: sentences coming from the same parent document should be grouped together, and the most coherent ordering of a group of sentences is how they were presented in their parent document. Typically, it is logical that sentences drawn from the same parent document would be similar. Moreover, grammatically and syntactically, it is logical that the sentences may be structured so that maintaining an invariant ordering would augment human comprehension.

Specifically, the Majority Ordering algorithm groups sentences by their parent document and then orders the blocks by the ranking of the highest-ranked sentence in a block. Figure 4.5 illustrates the differences between Similarity Ordering, Majority Ordering, and Block Ordering. The color of each sentence unit indicates the document it was selected from, and the suffix indicates the relevance score of that unit within the document.

3. Block Ordering: Intuitively, the Block Ordering algorithm is an amalgamation of the Similarity Ordering and Majority Ordering algorithms. The Block Ordering algorithm has two primary components. The first component involves grouping the sentences into blocks based on their parent document. This step is shared between the Block Ordering algorithm and the Majority Ordering algorithm. The second step involves ordering the grouped blocks of text.

The algorithm for ordering the blocks of texts combines document heuristics with our Similarity Ordering algorithm. We first order the blocks by their length (the number of sentences in the block). For blocks of equal length, we calculate the similarity of each block with the last fixed sentence. Hence, given the last sentence of the preceding block, we select the next block first by its length and then by the similarity of the block with the preceding sentence. If there is no single longest block to begin the answer, we select the longest block most similar to the full answer. This algorithm is tuned for specific goals with respect to human comprehension and readability. Grouping the sentences into blocks is done to maximize local coherence. The use of block length as an ordering heuristic is done to order topics by relevance. Finally, ordering blocks of equal length by similarity to the preceding sentence is done to maximize sentence continuity and fluidity.

In Figure 4.5c the green block is ordered first because it is the longest. The blue block is ordered second because it has the highest similarity score with sentence 3.4. The yellow block is ordered third because it has a higher similarity with sentence 2.2, and the red block is thus last.

4.3.2 Analysis of Ordering

Quantitative Analysis: To evaluate our approaches, we performed a manual analysis of 100 different answers, ordered by each of our proposed ordering algorithms (see Table 4.4). We rate each ordering as ‘reasonable’ or ‘unreasonable’. Note that this rating does not pass judgment on the correctness of the answer since it is designed for comparative analysis at the module level (i.e., to compare ordering approaches rather than content selection).

Algorithm	Reasonable	Unreasonable
Baseline	59	41
Similarity Ordering	55	45
Majority Ordering	71	29
Block Ordering	75	25

TABLE 4.4: Manual evaluation of sentence ordering

Qualitative Analysis: Because sentence ordering in the baseline system is based solely on question-answer relevance, we identified two major issues: global coherence and local coherence.

The global coherence issue is generally a problem of layout and cohesiveness. An ideal answer would begin with a broad answer to the question and move into more specific details and any available evidence to support the answer. Further, an ideal answer should not be hopping back and forth between topics and should stick to one before moving on to another. The baseline system did a decent job of beginning with a broad answer to the question because the input sequence is ordered by its relevance score. However, after the first sentence, answers tended towards redundant information and divergent trains of thought.

The local coherence issue has more to do with the semantics of the sentence and grammatical restrictions of the language. For instance, language like ‘There was also’ should not appear as the first sentence in an answer because this makes no sense logically. Additionally, certain words like ‘Furthermore’ indicate that the content of the sentence is highly dependent on the content of the preceding sentence(s), and the baseline ordering approach frequently breaks this dependency.

1. Similarity Ordering: We found that the Similarity Ordering performed poorly; only 55 of 100 answers were deemed ‘reasonable’. We believe this is due to the high degree of similarity between the candidate sentences in our domain. Because the candidate sentences are so similar, the results of clustering are highly variant and appear to be almost arbitrary at times. All the sentences contain similar language and key phrases that make it challenging to create meaningful sub-clusters. Additionally, one of the biggest problems with our system is due to the sentences that began with phrases like ‘However’ and ‘Furthermore’ that place strict requirements on the content of the preceding sentence. This was particularly problematic for the Similarity Ordering algorithm, which has no mechanism for making sure that such sentences are placed logically with their dependent sentences. The Similarity Ordering algorithm does perform relatively well in creating logical groups of sentences that cut down on how often an answer is jumping from one topic to another. Additionally, these groups are ordered well,

beginning with the more general of the two and then finishing with specifics and a presentation of the supporting data. However, we note that the problems with local coherence greatly outweigh the strengths in global coherence since a good answer can still be coherent, even if the organization could be improved. Whereas if the local coherence is poor, then the answer becomes nonsensical.

2. Majority Ordering: The Majority Ordering algorithm proved to be a successful method for ordering sentences, where 71 out of 100 answers were deemed ‘reasonable’. The Majority Ordering displayed very strong local coherence, which confirms the hypothesis that sentences should likely be kept in their original ordering to maximize human readability and coherence.

However, this algorithm faced issues with global coherence. It produced answers that start with a relevant topic more often than not; however, after the initial block, it struggled to smoothly transition from one block to the next. This is consistent with expectations for the Majority Ordering algorithm. The block with the highest-rated sentence is ordered first, which explains why the first block is frequently the most topically relevant. After the initial block placement, however, the algorithm makes no explicit attempts to manage or smooth transitions between blocks. Compared with the other two algorithms, this is where the Majority Ordering algorithm displays its poorest performance. It performs strongly when ordering sentences within a block, enforcing local coherence so that sentences beginning with language such as ‘Finally’, ‘Lastly’, ‘Therefore’, etc., followed a related sentence that satisfied the sequential dependency.

3. Block Ordering: The Block Ordering algorithm produced the best answers, with 75 out of 100 answers ranked as ‘reasonable’. This is consistent with our expectations, as the Block Ordering algorithm effectively combines the most substantial aspects of the Majority Ordering and Similarity Ordering algorithms. With respect to local coherence, this algorithm displays similar performance when compared to the Majority Ordering algorithm while displaying stronger coherence between blocks (due to the use of a similarity metric to order blocks). This algorithm also showed the strongest global coherence, which is likely due to first grouping the sentences into blocks and then ordering them.

This algorithm displayed one core weakness, which is its inability to identify high-quality opening sentences. This is due to the usage of block length as a heuristic for topic relevance. While in most cases, this heuristic proved to be successful, accounting for these outliers may significantly improve the performance of the Block Ordering algorithm. We note that the Block ordering algorithm performed well in producing high-quality and coherent answers. We can see that Block Ordering performs the best with respect to the simple coherence evaluation we conducted.

Modeling Approach - Sentence Fusion: An observed weakness of the original system is that the generated summaries often contain highly repetitive information. While MMR is added in the pipeline to deal with redundancy and maximize the diversity of covered information, extractive summarization still picks entire sentences that may partially overlap with

a previously selected sentence. To tackle this problem, we introduce sentence fusion to identify common information among sentences and apply simple abstractive techniques over the baseline extractive summaries.

Methodology: Given a set of candidate sentences generated by the pipeline for each summary, the sentence fusion module operates in two steps: 1) the candidate set is expanded to include fused sentences, and 2) sentences are selected from the expanded set to produce a new summary.

Expansion of Candidate Set: To generate fused sentences, we begin by building upon previous work on multiple-sentence compression (Filippova, 2010), in which a directed word graph is used to express sentence structures. The word graph is constructed by iteratively adding candidate sentences. All words in the first sentence are added to the graph by creating a sequence of word nodes. A word in the following sentence is then mapped onto an existing word node if and only if it is the same word, with the same part of speech. Our assumption is that a shared node in the word graph is likely to refer to the same entity or event across sentences.

We then find a K-possible fused sentence by searching for the K-shortest path within the word graph. Definition of the edge weights follows from work by Filippova (2010):

$$w(e_{ij}) = \frac{freq(i) + freq(j)}{\sum_{s \in S} diff(s, i, j)^{-1}} \times freq(i) \times freq(j)$$

where $diff(s, i, j)$ is the difference between the offset positions of word i and j in sentence s . Intuitively, we want to promote a connection between two word nodes with close distance and between nodes that have multiple paths between them. We also prefer a compression path that goes through the most frequent no-stop nodes to emphasize important words.

When applying the sentence fusion technique to the BioASQ task, we first pre-process the candidate sentences to remove transition words like ‘Therefore’ and ‘Finally’. Such transition words may be problematic because they are not necessarily suitable for the new logical intent in fused sentences and may break the coherence of the final answer. We also constrain fusion so that the fused sentences are more readable. For instance, we only allow the fusing of pairs of sentences that are of proper length to avoid generating overly complicated sentences. We also avoid fusing sentences that are too similar or too dissimilar. In the first case, information in the two sentences is largely repetitive, so we simply discard the one containing less information. In the latter case, fusing two dissimilar sentences more likely confuses the reader with too much information rather than improving the sentence readability. Finally, we add a filter to discard ill-formed sentences, according to some hand-crafted heuristics.

Selecting Sentences from Candidate Set: The next step is to select sentences from the candidate set and produce a new summary. An Integer Linear Program (ILP) problem is formulated as follows, according to Gillick and Favre (2009):

	Method	R-2	R-SU4	Avg Pr.	Avg Re.	Avg F1	Avg Len
1	Baseline System	0.6948	0.6890	0.2297	0.8688	0.3207	173.31
2	MMR + Order	0.6291	0.6197	0.2758	0.8118	0.3633	140.39
3	MMR + Fusion	0.6183	0.6169	0.2783	0.8094	0.3687	139.24
4	MMR + Relevance + Order	0.6357	0.6256	0.2728	0.8124	0.3606	143.55
5	MMR + Relevance + Order + Post	0.6215	0.6126	0.2788	0.8111	0.3668	139.10
6	MMR + Relevance + Order + Fusion + LM	0.6114	0.6042	0.2775	0.8113	0.3682	141.21
7	MMR + Relevance + Order + Fusion	0.6213	0.6101	0.2686	0.8099	0.3579	143.94
8	MMR + Relevance + Order + Fusion + Post	0.6017	0.5932	0.2775	0.8091	0.3653	140.38
9	MMR + Fusion + Order	0.6223	0.6159	0.2840	0.8181	0.3745	138.79
10	MMR + Fusion + Relevance + Order	0.6257	0.6214	0.2825	0.8193	0.3730	139.73
11	MMR + Fusion + Relevance + Order + Post	0.6149	0.6096	0.2886	0.8126	0.3768	136.43
12	Fusion + MMR + Relevance + Order	0.6112	0.6103	0.2837	0.8211	0.3723	142.11
13	Fusion + MMR + Relevance + Order + Post	0.6048	0.6040	0.2898	0.8143	0.3789	137.78

TABLE 4.5: Performance of different module combinations on Test Batch 4, BioASQ 4th edition; R=Rouge, Pr=Precision, Re=Recall

$$\max_{y,z} \sum_{i=1}^N w_i z_i, \text{ such that } \sum_{j=1}^M A_{ij} y_j \geq z_i, A_{ij} y_j \leq z_i,$$

$$\sum_{j=1}^M l_j y_j \leq L, y_j \in \{0, 1\}, z_i \in \{0, 1\}$$

In the equation, z_i is an indicator of whether concept i is selected into the final summary, and w_i is the corresponding weight for the concept. The goal is to maximize the coverage of important concepts in a summary. We assign diminishing weights during the actual experiments so that later occurrences of an existing concept are less important. This forces the system to select a more diverse set of concepts. We follow the convention of using bigrams as a surrogate for concepts (Berg-Kirkpatrick et al., 2011; Gillick et al., 2008), and bigram counts as initial weights. Variable A_{ij} indicates whether concept i appears in sentence j , and variable y_j indicates if a sentence j is selected or not.

4.3.3 Experiments and Results

4.3.3.1 Quantitative Analysis

Table 4.5 shows the results of different configurations of the ordering and fusion algorithms (Rows 1 - 4, Row 7, Row 9). Though the overall ROUGE score drops slightly from 0.69 to 0.61 after sentence fusion with the ILP-selection step, this is still competitive with other systems (including the baseline). The sentence re-ordering does not directly impact the ROUGE scores.

4.3.3.2 Qualitative Analysis

We manually examined the fused sentences for 50 questions. We found that our sentence fusion technique is capable of breaking down long sentences into independent pieces and is,

therefore, able to disregard irrelevant information. For example, given a summary containing the original sentence:

‘Thus, miR-155 contributes to Th17 cell function by suppressing the inhibitory effects of Jarid2. (2014) bring microRNAs and chromatin together by showing how activation-induced miR-155 targets the chromatin protein Jarid2 to regulate proinflammatory cytokine production in T helper 17 cells’.

Our fusion technique is able to extract important information and formulate it into complete sentences, producing a new summary containing the following sentence:

‘Mir-155 targets the chromatin protein jarid2 to regulate proinflammatory cytokine expression in th17 cells’.

The fusion module is also able to compress multiple sentences into one, with minor grammatical errors. For example:

Sentence 1: *‘The RESID Database is a comprehensive collection of annotations and structures for protein post-translational modifications including N-terminal, C-terminal and peptide chain cross-link modifications[1].’*

Sentence 2: *‘The RESID Database contains supplemental information on post-translational modifications for the standardized annotations appearing in the PIR-International Protein Sequence Database[2].’*

Our approach produces the fused sentence:

‘The RESID Database contains supplemental information on post-translational modifications[1] is a comprehensive collection of annotations and structures for protein post-translational modifications including N-terminal, C-terminal and peptide chain cross-link modifications[2].’

However, the overall quality of fused sentences is not stable. As shown in Figure 4.6, around 25% of the selected sentences in final summaries are fused. Among the fused sentences, 47% improved the overall readability by reducing redundancy and repetition. 5% of the sentences have improved readability with minor grammatical errors, such as a missing subordinate conjunction or superfluous discourse markers. 8% of the fused sentences did have an appreciable effect on readability. However, a large number of fused sentences (around 26 %) were not coherent and degraded the quality of the answer.

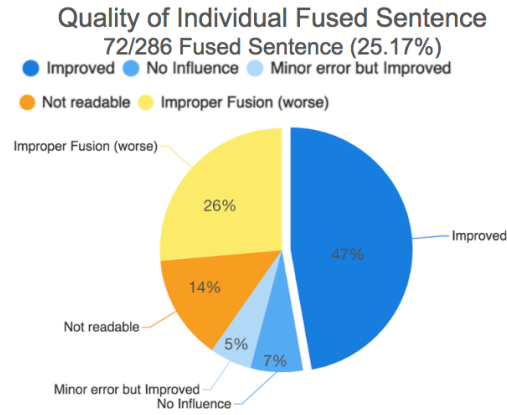


FIGURE 4.6: Quality of Fused Sentences

Further Improvements: In order to further improve the performance of our system, we made a few modifications to each module in the system and improved the overall architecture of the module pipeline:

- **Modification of System Architecture:** We intuited that the ILP process in the sentence fusion model could not handle a very large number of candidate inputs, producing a lot of (redundant, similar) fused sentences. To resolve this problem, we removed the ILP model from the sentence fusion step. We also moved the sentence fusion step before the sentence selection module (Rows 12-13) so that the MMR algorithm in the sentence selection module could eliminate redundant fused sentences.
- **Modifications to Sentence Selection Module and Relevance Ranker:** For the sentence selection module, we modified the original MMR model. The original MMR model selected a fixed number of sentences, which naturally introduced repetition. In order to reduce repetition, we built a so-called ‘*Early-Stop MMR*’, which stops selecting sentences when the maximum overlap score grows beyond a certain threshold, and the minimum relevance score drops down below another threshold (Rows 4-8).

For the relevance ranker, we explore an alternative similarity metric (Row 6). The Query Likelihood Language Model (Schütze et al., 2008) is widely used in information retrieval. We formulated the relevance ranking procedure as an information retrieval problem and used a language model so that long sentences would get a higher penalty.

- **Post-Processing:** To further reduce repetition, we add an additional filter before final concatenation by iteratively adding the selected sentences to the final output, and discarding a sentence if it is too similar to the existing summary (Rows 8,11 and 13).

Analysis - Effects of Individual Modules: Table 4.5 shows the results of extensions to the baseline system. Two systems are highlighted (Rows 5 and 10), as they give the most balanced results between the quality of retrieved information and conciseness: one system performs sentence selection, then ranks sentences before ordering by relevance, and applies

	Question	Which syndrome is associated with mutant DVL1?
	Ideal Answer	Mutations in DVL1 cause an osteosclerotic form of Robinow syndrome.
1	MMR + Relevance + Order + Fusion	We identified de novo frameshift mutations in DVL1, a mediator of both canonical and non-canonical Wnt signaling, as the cause of RS-OS, an RS subtype involving osteosclerosis, in three unrelated individuals. Argeted Sanger sequencing in additional subjects with DRS uncovered DVL1 exon 14 mutations in five individuals, including a pair of monozygotic twins. <u>DVL1 frameshift mutations clustering in the penultimate exon cause autosomal-dominant Robinow syndrome.</u> Mutations in DVL1 cause an osteosclerotic form of Robinow syndrome.
2	MMR + Relevance + Fusion + Order	Mutations in DVL1 cause an osteosclerotic form of Robinow syndrome. <u>DVL1 frameshift mutations clustering in the penultimate exon cause autosomal-dominant Robinow syndrome.</u> We identified de novo frameshift mutations in DVL1, a mediator of both canonical and non-canonical Wnt signaling, as the cause of RS-OS, an RS subtype involving osteosclerosis, in three unrelated individuals. Argeted Sanger sequencing in additional subjects with DRS uncovered DVL1 exon 14 mutations in five individuals, including a pair of monozygotic twins.
3	Fusion + MMR + Relevance + Order + Post	DVL1 frameshift mutations in DVL1 cause an osteosclerotic form of Robinow syndrome. We identified de novo frameshift mutations in DVL1, a mediator of both canonical and non-canonical Wnt signaling, as the cause of RS-OS, an RS subtype involving osteosclerosis, in three unrelated individuals. Argeted Sanger sequencing in additional subjects with DRS uncovered DVL1 exon 14 mutations in five individuals, including a pair of monozygotic twins.

TABLE 4.6: System performance comparing Fusion + Ordering and Ordering + Fusion

the additional post-processing step (Row 5); the other system performs sentence selection, fusion, and then ranks sentences prior to ordering without the post-processing step (Row 10).

Rows 5, 8, 11, and 13 show the effectiveness of the additional post-processing step. Overall, this procedure is able to reduce the answer length while preserving important information. We observed that the post-processing step is less effective when fusion is performed after MMR. This is because, in these settings, there is an additional sentence selection step in the fusion module using integer linear programming that forces the selected sentences to be diverse. In all other settings, including when fusion is performed prior to MMR, we only have one sentence selection step. Since MMR iteratively selects sentences according to both similarity and relevance, the last selected ones may be informative but repetitive. Row 6 shows our experiments with language modeling; the language model gives a higher penalty to longer sentences, producing shorter but less informative results.

Analysis - Impact of System Architecture: Exploring the performance of systems using different architectures, we observed that systems with fusion prior to ordering could generate more logically coherent summaries. Table 4.6 shows an example. All underlined sentences express the same fact that DVL1 is the cause of Robinow syndrome. In Row 1, where fusion is performed after ordering, there is a sentence that serves as an explanation between the underlined sentences, which breaks the logical coherence. In Row 2 and Row 3, where ordering is performed after fusion, the generated answers demonstrate better coherence: All underlined

sentences are placed together, followed by the explanation; The opening sentences are also more concise and more directly related to the question.

We also experimented with architectures where the fusion module is run prior to MMR, and MMR is used as the only sentence selection step. In these systems, MMR receives many fused sentences that overlap and complement each other at the same time because all similar sentences are fused prior to sentence selection. As a result, such architectures sometimes produce summaries that are more repetitive compared to others.

4.4 Conclusions and Prospective Future Directions

An effective narrative not only talks about the right content but organizes the content in a fairly coherent manner. This forms a structural layout. In this chapter, we have explored N-Local and N-Global anchoring into the structure in order to generate a narrative. Representation of structural format is not readily tangible and thereby not trivially interpretable. Hence we make use of clustering and reordering techniques to tackle this problem.

For N-Local anchoring, we looked into the visual and text domains where the task is to generate textual step-wise instruction for a *'how-to'* activity such as cooking recipes. The focus here is instilling structure learnt from FSMs in neural models for sequential procedural text generation with multimodal data. We gather a dataset of 16k recipes where each step has text and associated images. We propose two ways of imposing structure from phases and states of a recipe derived from FSM N-locally for each step. The first model imposes structure on the decoder and the second model on the loss function by modeling it as a hierarchical multi-task learning problem. We show that our proposed approach improves upon the baseline and achieves a METEOR score of 0.31.

Similarly, we revisit query-based summarization to address N-Global anchoring in structure. Though extractive summarization techniques can be developed to maximize performance as measured by evaluation metrics like ROUGE, such systems suffer from human readability issues, as mentioned above. In this chapter, we attempted to combine extractive techniques with simple abstractive extensions by extracting the most relevant non-redundant sentences, re-ordering and fusing them to make the resulting text more human-readable and coherent. Using an initial set of 100 candidate answer sets, we experimented with different ordering algorithms such as Similarity, Majority, and Block Ordering. We identified that Block Ordering performs better the others in terms of global and local coherence. We then introduced an Integer Linear Programming based fusion module that is capable of not only fusing repeated content but also breaking down complicated sentences into simpler sentences, thus improving human readability. The improved baseline system achieved a ROUGE-2 of 0.6257 and ROUGE-SU4 of 0.6214 on test batch 4 of BioASQ 4b.

Based on the takeaways learnt from this chapter, I believe the following directions are worth investing our efforts to bring significant improvements in the way structure is anchored in narratives.

1. Interpretability of Latent Structure: The N-Local anchoring relies on the sequential input of structure anchors at each and every step. This structural representation is provided in the form of phase and state sequences from clustering and state machines, respectively. We have performed a grid search over the number of phases and states in this process. Interpreting each of these may provide more intuition into the semantic structuring of the narrative. This also leads to an immediate intermediary step of evaluating the latent structure unit explicitly.

2. Backpropable Scaffold for Anchors: We have explored an external memory mechanism that acts as a scaffold to structure. Investigating backpropable variants as a scaffold for structure such as hidden markov models are worth investing effort in. This also is a prospective direction to pave the way into dynamic planning.

3. Partial Anchoring: As discussed in the previous chapter, the idea of this point is similar. In this chapter, we explored the ends of the spectrum based on anchoring each unit explicitly and at the overall narrative level. There are intermediate techniques such as dynamic planning that rely on generating a whole component of one structural unit instead of each unit of the narrative. This direction is promising as we can hypothesize improved local coherence with respect to the structural unit.

4. Non-autoregressive sentence order prediction: An overall sentence ordering is performed by keeping the entire narrative in mind. While this takes care of relative presence, another possibility that can be extended to neural models is the prediction of sentence orderings non-autoregressively. In this way, the mini-structure within a structure is partly agnostic to long-distance contexts, thereby giving more importance to surrounding contexts. In addition, if ordering is performed at the end of the network as a post-editing step, non-autoregressive techniques would enable parallel computations, thereby speeding up the process, which is particularly helpful in real-time systems.

5

Surface Form Realization: N-Local and N-Global Anchoring

To bring relevance to people, you have to be able to speak their language effectively.

Sunday Adelaja

Relevance in itself is not sufficient, it is also important to realize or present it in varied ways depending on personalities or languages. So far, we have looked into determining anchoring in the right content and organizing it by anchoring in the appropriate structure. The obvious next step is in the choice of surface form words that pose a challenge in how these sentences are generated. Hence, we move to our next step, which is anchoring the generation in surface form realization. This is often referred to in the literature as stylistic variation (Schilling, 2013; Harrison et al., 2019). One such variation in styles is reflected through various personas, which impact the way a story is narrated, as studied by Mairesse and Walker (2010). In this chapter, we are going to look into surface form realization in two forms. In the first part, I will present surface form choices from different languages, and in the second part, we will discuss more traditional definitions with respect to persona and style.

Part of this chapter is based on the following papers:

- “My Way of Telling a Story”: *Persona based Grounded Story Generation* (Chandu et al., 2019c)
- “Style Variation as a Vantage Point for Code-Switching” (Chandu and Black, 2020b)
- “Language informed modeling of code-switched text” (Chandu et al., 2018)
- “Speech Synthesis for Mixed-Language Navigation Instructions” (Chandu et al., 2017b)

In this section, our main focus is going to be on surface realization when multiple languages interact with one another within a single utterance. This phenomenon is also known as code-switching (Milroy et al., 1995). Code-Switching (CS) is a widely studied linguistic phenomenon where two different languages are interleaved. This occurs within multilingual communities (Poplack, 1980; Myers-Scotton, 1997; Muysken, 2000; Bullock and Toribio, 2009). Typically one language (the *matrix language*) provides the grammatical structure for CS text, and words are inserted from another language (the *embedded language*). Different languages are varied surface form realizations of the same content. Having said this, we do not understate the obvious forms of differences in concepts perceived through the lens of languages (Yaguello et al., 1998; Deutscher, 2010). We are assuming to deal with concepts that are generic across languages. There is no strict constraint on the number of languages that participate in the interaction. However, typically this intermixing is between two languages. In the case of code-switching for this chapter, we will be looking into the mixing of two different languages. The choice of selecting a language to express each word is the choice we have in the forms of ‘*surface realization*. The obvious guidance that can be provided in this scenario is via language id.

This language id information can be used both as N-Local and N-Global anchors while dealing with code-switched text. In the case of N-Local anchoring, the language id for each lexical unit is provided separately and distinctly. We observe how this way of N-Local anchoring of language ids helps in training a code-switched language model. We will also explore the usage of identifying the language of each lexical unit in synthesizing speech instructions with entities belonging to more than one language. Here, each of the lexical units is directly guided through an anchor unit α_i^j which is the language id. Note that since the guidance is provided at the word level, the output unit is also a word instead of a sentence anchoring the content and structure.

In most cases, gathering annotated code-switched data is a challenge for several language pairs. Even if we manage to find this data in the wild, annotating it with lexical level language ids is a very expensive task and manually intensive task. Automatically identifying the language ids is still an active area of research (Molina et al., 2016). However, we have access to monolingual corpora for many languages, except a long tail of low-resource languages. We plan on making use of these monolingual resources to provide overall guidance from the language id information in generating code-switched text. The language id information is not explicitly given to each lexical unit, but the representation in the latent space is modeled to discriminate the levels of mixing between languages.

5.1 Related Work

Overview of Modeling Code-Switched text: Early efforts to develop computational framework for CS data include Joshi (1982) in the 1980s and Goyal et al. (2003); Sinha and Thakur (2005); Solorio and Liu (2008a,b) in the 2000s. However, these methods are quite limited in their applicability to the kind of data we see on the Internet and social media where code-switching data is usually found. This has led to an increased attention of the NLP community in the areas of Language Modeling (Li and Fung, 2013, 2014; Adel et al., 2015, 2013a,b; Garg et al., 2017),

POS tagging (Vyas et al., 2014; Jamatia et al., 2015; Çetinoğlu and Çöltekin, 2016), language identification (King and Abney, 2013) prediction of code-switch points (Das and Gambäck, 2014), sentiment analysis (Rudra et al., 2016) and also certain meta level studies that include understanding metrics to characterize code-mixing (Patro et al., 2017; Guzmán et al., 2017).

Code-Switched Language Models: Neural Language Models have a limitation in capturing only a finite context. This was overcome in Recurrent Neural Network (RNN) based Language Model (Mikolov et al., 2010). The RNN has an input layer x , hidden layer s (also called context layer or state), and an output layer y . What allows for capturing of an infinite context is that the input to the network at time t , i.e., $x(t)$ is concatenated with $s(t - 1)$, i.e., the output of the hidden layer at time $t - 1$ to compute the current state $s(t)$ and output $y(t)$ which is the probability of the next word given the context. Several word-level tasks can be tackled better at the constituent character-level (Zhang et al., 2015; Chung et al., 2016). Character-Level Language Models (Kim et al., 2016) represent a word w as a collection of the embeddings of the constituent characters $\mathbb{C}^w = [c_1, c_2, \dots, c_l]$ where l is the length of w . \mathbb{C}^w is then passed through a Convolution Neural Network (CNN) with varying filter widths to capture different lengths of character n-grams. Essentially, if there are h filters, then the feature mapping for w is $y^w = [y_1, y_2, \dots, y_h]$. This feature mapping instead of word embeddings is fed to the RNN-LM as described above, and the rest of the procedure remains the same.

There has been some recent focus on adapting existing language models for CS text. Li and Fung (2013, 2014) use a translation model together with the language model of the matrix language to model the mixed language. Linguistic features reduce the search space within the translation model in CS texts like inversion constraint and functional head constraint (Sankoff and Poplack, 1981).

N-Local Anchoring from Language Modeling: In another approach, Adel et al. (2015), use a Factored Language Model (FLM) that includes syntactic and semantic features found in CS text that are indicative of a switch e.g., trigger words, trigger POS tags, brown cluster of function and content words that result in a significant reduction in perplexity. Another recent method called Dual Language Model (DLM) (Garg et al., 2017) combines two monolingual language models by introducing a ‘switch’ token common to both languages. Predicting this word in either language acts as a proxy to the probability of a switch, and the next word is then predicted using the LM of the language that was switched to. Among neural methods, Adel et al. (2013a) use an RNN-based LM to predict the language of the next word along with the actual word to model CS text. Following on these intuitions, our models are built on top of the AWD-LSTM LM (Merity et al., 2017) that was chosen due to its accessibility and high performance (recently State of the Art) on the Penn-Tree Bank and Wikitext-2 dataset (Merity et al., 2016). Extensive work has been done on this model through investigation on the relative importance of hyper-parameters (Merity et al., 2018).

A natural extension to language modeling is the generation of text. There have been several studies that relied on constraint-theory-based generation techniques. Li and Fung (2012) combined syntactic constraints by predicting language boundary to reconstruct CS text. Pratapa et al. (2018) and Lee et al. (2019) present techniques based on Equivalence Theory (Poplack,

1980) and Matrix Language Frame Theory (Myers-Scotton, 1997) to create grammatically valid CS text. While these methods demonstrate the use of expert knowledge to assist generation, the same is difficult to replicate and scale to other languages.

Prior works rely on annotations of language spans in multi-task setup (Chandu et al., 2018) or using dual RNN to handle each language (Garg et al., 2018b). Capturing syntactic and language switching signals proves effective in a hierarchical VAE architecture (Samanta et al., 2019).

Anchoring Language in Speech Systems: Previous work in synthesizing multilingual speech can be classified into three approaches: bilingual TTS systems in which two speech databases are used from the same speaker to build a single TTS system, polyglot systems that create combined phonesets, and phone-mapping based approaches. Bilingual TTS systems have been proposed by Liang et al. (2007) for English-Mandarin code switched TTS. Microsoft Mulan (Chu et al., 2003) is a bilingual system for English-Mandarin that uses different frontends to process text in different languages and then uses a single voice to synthesize the text. Both these systems synthesize speech using native scripts; that is, each language is written using its own script. Polyglot systems (Traber et al., 1999) enable multilingual speech synthesis using a single TTS system. This method involves recording a multi-language speech corpus by someone fluent in multiple languages. This speech corpus is then used to build a multilingual TTS system. The primary issue with polyglot speech synthesis is that it requires the development of a combined phoneset, incorporating phones from all the languages under consideration.

Another type of multilingual synthesis is based on phone mapping, whereby the phones of the foreign language are substituted with the closest sounding phones of the primary language. This method results in a strong foreign accent while synthesizing the foreign words, which may or may not be acceptable. Also, if the sequence of the mapped phones does not exist or does not frequently occur in the primary language, the synthesis quality can be poor. To overcome this, an average polyglot synthesis technique using HMM-based synthesis and speaker adaptation has been proposed (Latorre et al., 2006). Such methods make use of speech data from different languages and speakers.

A framework for speech synthesis of code-mixed text was proposed by Sitaram and Black (2016); Sitaram et al. in which we assumed that two languages were mixed, and one of the languages was not written in its native script but borrowed the script of the other language. This framework consisted of first identifying the language of a word using a dictionary or HMM-based approach, then normalizing spellings of the language that was not written in its native script, and then transliterating it from the borrowed script to the native script. Then, we used a mapping between the phonemes of both languages to synthesize the text using a TTS system trained on a single language. We performed experiments on German-English and Hindi-English. We also conducted experiments to determine which language's TTS database should be used when synthesizing code-mixed text.

In this chapter, we extend on this previous work in two ways: (1) Our current system is a bilingual system built using speech from two monolingual speech datasets and a combined phoneset, thereby removing the need for a phone to phone mapping (2) We formulate our proposed approach and determine its effectiveness in the domain of navigation instructions. The synthesis depends on lexical level language anchors.

N-Global Anchoring in Code-Switching: As we have seen in the sub-topic of N-Local anchoring of language id for code-switched generation, most of the prior work falls under this domain. [Chang et al. \(2018b\)](#) proposed a GAN-based approach to generate language id tags and discriminate whether it is a valid sequence. This process internally uses an explicit representation of language tags though this is not provided as a part of the input. [Winata et al. \(2019\)](#) proposed a seq-to-seq model with a copy mechanism limiting the method to rely on parallel monolingual translations of CS text.

Our approach for N-Global anchoring the language information eliminates the need for drafting constraint theories, additional annotations for language ids, and parallel data. This enables scalability to new language pairs attributed to the availability of monolingual corpora and limited CS text.

Emotion in Text Generation: [Cavazza et al. \(2009\)](#) have stressed the importance of expressing emotions in the believability of the automated storytelling system. Adapting a personality trait hence becomes crucial to capture and maintain the interest of the audience. Associating the narrative to a personality instigates a sense of empathy and relatedness. Although there has been research in generating persona-based dialog responses and generating stylistic sentences ([Shuster et al., 2018](#); [Fu et al., 2018](#); [Prabhumoye et al., 2018](#); [Shen et al., 2017a](#)), generating persona-based stories with different personality types narrating them has been unexplored. In this chapter, we focus on generating a story from a sequence of images as if the agent belongs to a particular personality type.

Style Transfer: One line of research that is closely related to our task is style transfer in text. Recently generative models have gained popularity in attempting to solve style transfer in text with non-parallel data ([Hu et al., 2017](#); [Shen et al., 2017a](#); [Li et al., 2018a](#)). Some of this work has also focused on transferring author attributes ([Prabhumoye et al., 2018](#)), transferring multiple attributes ([Lample et al., 2019](#); [Logeswaran et al., 2018](#)) and collecting parallel dataset for formality ([Rao and Tetreault, 2018](#)). Although our work can be viewed as another facet of style transfer, we have strong grounding of the stories in the sequence of images.

Persona Based Dialog: Persona-based generation of responses has been studied by the NLP community in the dialog domain. [Li et al. \(2016b\)](#) encoded personas of individuals in contextualized embeddings that capture the background information and style to maintain consistency in the responses given. The embeddings for the speaker information are learnt jointly with the word embeddings. Following this work, [Zhou et al. \(2018c\)](#) proposed Emotional Chatting Machine that generates responses in an emotional tone in addition to conditioning the content. The key difference between former and latter work is that the latter captures the dynamic change in emotion as the conversation proceeds, while the user persona remains the same in the former case. [Zhang et al. \(2018\)](#) release a huge dataset of conversations conditioned on the persona of the two people interacting. This work shows that conditioning on the profile information improves the dialogues, which is measured by the next utterance prediction. In these works, the gold value of the target response was known. For our work, we do not have

gold values of stories in different personas. Hence we leverage annotated data from a different task and transfer that knowledge to steer our generation process.

Multimodal domain: With the interplay between visual and textual modalities, an obvious downstream application for persona-based text generation is image captioning. [Chandrasekaran et al. \(2018\)](#) worked on generating witty captions for images by both retrieving and generating with an encoder-decoder architecture. This work used external resources to gather a list of words that are related to puns from the web, which the decoder attempts to generate conditioned on phonological similarity. [Wang and Wen \(2015\)](#) studied the statistical correlation of words associated with specific memes. These ideas have also recently penetrated into the visual dialog setting. [Shuster et al. \(2018\)](#) have collected a grounded conversational dataset with 202k dialogs where humans are asked to portray a personality in the collection process. They have also set up various baselines with different techniques to fuse the modalities, including multimodal sum combiner and multimodal attention combiner. We use this dataset to learn personas which in turn are adapted to our storytelling model.

5.2 N-Local Anchoring in Language Information

As we have discussed before, N-Local anchoring provides definitive and specific categorical grounding while modeling text. Our main focus is on utilizing these fine-grained anchoring units in generation. We use existing technologies to determine these anchors. In this case, we will be using lexical level language id information in modeling text. In §5.2.1, we will be looking into building a language model for code-switched text using the anchors at lexical units along with the input and predicting the same explicit anchor units in a multi-task learning framework. Following this in §5.2.2, we will delve into utilizing these lexical level language id units in improving the quality of mixed-language instructions.

5.2.1 N-Local Anchoring for Language Modeling

We approach Code-Switching through Language Modeling (LM) on a corpus of Hinglish (Hindi + English) that we collected from blogging websites containing 59,189 unique sentences. We implement and discuss different Language Models derived from a multi-layered LSTM architecture. Our main hypothesis is that providing language id information explicitly for each individual word builds a robust language model as opposed to simple word-level models by learning the switching points. We attempt this in two ways: (1) factored model learning embeddings both for input word and input language, and (2) multi-task learning of predicting the language of the next word along with the word itself. We show that our highest performing model achieves a test perplexity of 19.52 on the CS corpus that we collected and processed. On this data, we demonstrate that our performance is an improvement over AWD-LSTM LM (a recent State of the Art on monolingual English).

Code-Switched data is quite challenging to obtain as this phenomenon is usually observed in informal settings. Moreover, data obtained from online sources are often noisy because of

spelling, script, morphological, and grammatical variations. These sources of noise make it quite challenging to build robust NLP tools (Çetinoğlu et al., 2016). Our goal is to improve LM for Hindi-English code-mixed data (*Hinglish*) where similar challenges are apparent. The task of language modeling is very important to several downstream applications in NLP including speech recognition, machine translation, etc. This task also aids in multi-task learning where parameters are shared between the language model and the other task. This is particularly important in domains such as code-switching that lack annotated data, where the necessity to leverage unsupervised techniques or transfer the representations from an unsupervised or self-supervised setup is crucial. In this section, we present our attempt to this problem using two techniques: (1) Factored language model that contains the factors of word and language given to a stacked LSTM based architecture, and (2) Posing this as a multi-task learning (MTL) problem with a dual objective including predicting the next word, and predicting the language of the next word. Learning to predict the language of the next word allows the model to switch points between languages at a global level. In the path towards building the MTL architecture, we attempted to explicitly learn the language of the current word without predicting the language of the next word. This is one of the models in our ablation studies, which is similar to and is inspired from the factored LM.

In addition to the techniques used for monolingual language modeling, providing information about the language is a key component in a code-switched domain. Our main goal in this section is to examine the effect of language information in modeling code-switched text. We approach this systematically by experimenting with ablations of encoding and decoding language id along with the word itself. In this way, the model implicitly learns the switch points between the languages. We achieve the least perplexity score in combination with a language-informed encoder and a language-informed decoder among the ablation of multiple models.

5.2.1.1 Data Collection and Description

To the best of our knowledge, there was no standard dataset to evaluate LM in code-switched texts at the time this work was done. In this subsection, we will describe our data collection and provide a brief analysis of it.

Data Collection: Curating a reasonable dataset for code-switched text is an important challenge for researchers in this domain. To the knowledge of the authors, there was no benchmark code-switched corpus for language modeling as there is for English (Merity et al., 2016; Marcus et al., 1994). Code-Switching is commonly observed in informal settings and in casual conversations. Hence the two potential source choices to gather data include social media (such as Twitter and Facebook) and blogging websites. We decided to go with the latter due to comparatively lesser noise and the availability of more descriptive text. Our data for code-switched

Criteria	Train	Dev	Test
# Sentences	35513	11839	11837
Avg Length of Sentences	18.90	17.58	18.22
Multilingual Index	0.8892	0.8905	0.8914
Language Entropy	0.6635	0.6639	0.6641
Integration Index	0.3304	0.3314	0.3312
Unique Unigrams	35,769	18,053	19,330
Unique Bigrams	276,552	125,108	130,947
Unique Trigrams	553,866	219,098	229,967

TABLE 5.1: Hinglish Data Statistics for Code-Switched Language Modeling

language modeling was collected after having crawled eight blogging Hinglish websites¹ that were returned by popular search engines (such as Google and Bing) with simple code-switched queries in the domains of health and technology. These code-switched texts cover several different topics, primarily technical reviews of electronic and general e-commerce products as well as several health-related articles. These texts were all tokenized at the sentence level, and lexical language identification was performed. All the sentences that did not have at least one word each from both languages were discarded to channel our problem towards tackling intra-sentential code-switching. This resulted in a total of 59,189 unique sentences. The data needed extensive cleaning due to a lot of hyperlinked text spans in varying formats.

Pre-processing: One of the important characteristics of code-switched text in Hinglish, when written in Roman script, is the non-standard representations of the words. This is very commonly observed since there are strict guidelines that monitor the correctness of an approximately phonetically represented word. There is no standard one-to-one correspondence of syllables in Hindi when written in the Roman script that universally everyone would follow in informal settings. Hence the same word could be written in multiple representations based on the idiosyncrasies of the individual and perception of Romanization of a syllable to that specific person.

For models encompassing language informed encoders, we perform lexical language identification for each of the words and annotate them with this information.

The non-standardized representations of the words are dealt in the following two ways:

- Soundex Encodings: We use *soundex encodings* of the words as another factor. This way, the idiolectic representational variations with an additional ‘h’ for aspirated sounds and single vs. multiple vowels for long and short syllables etc., would be mapped to the same space. This is not a solution to normalizing the representational variations, but

¹Hinglish blogging websites:
www.hinglishpedia.com
www.queshiinfotech.com
www.hindimehelp.com
www.pakkasolutionhindi.com
www.myhelplive.blogspot.com
www.seekhoweb.com
www.onlinesikhe.com

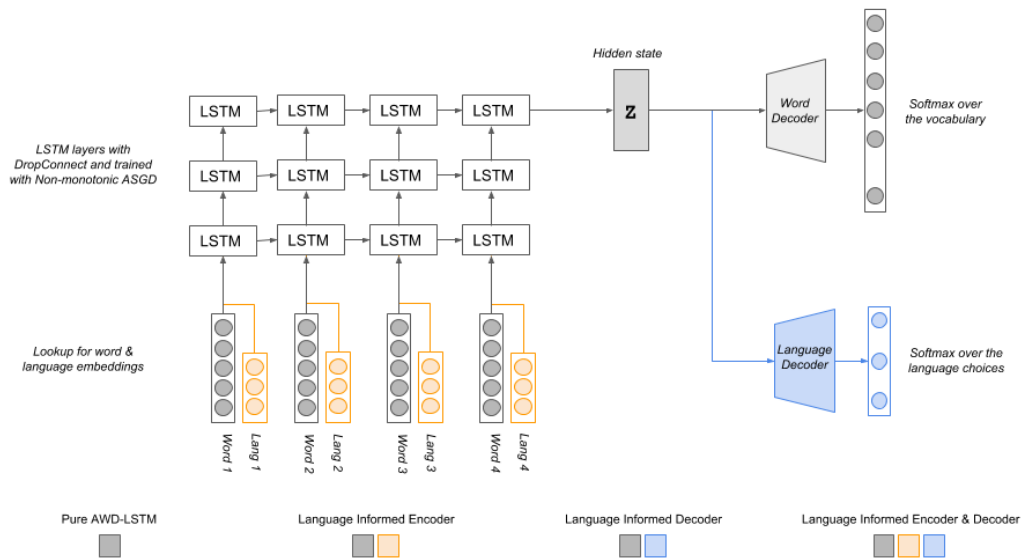


FIGURE 5.1: Anchoring language id units in Language Modeling of Code-Switching

we hypothesize that this feature helps learn context around variants of the same word better.

- **Transliteration:** Along the similar lines of standardizing the multiple representations of the same word, the words identified as Hindi at the lexical level are transliterated into their Devanagari representation, and the most likely Devanagari spelling in a pre-existing Devanagari dictionary is chosen according to soundex encoding.

Data Analysis: To estimate the quality and extent of mixing and frequency of switching in our data, we measured the Multilingual index (M-Index), Language Entropy and Integration index (I-index) that were introduced in the domain of CS by [Guzmán et al. \(2017\)](#). A multilingual index of 1 indicates an equal extent of mixing from both the participating languages. As we can observe, the mixing is nearly in the range of 0.8, which indicates that both Hindi and English are participating in the ratio of 4 is to 5, respectively. The metric itself does not reveal which is the embedded language and which is the matrix language. These metrics, along with other n-gram statistics over our data, are presented in Table 5.1. Note that these code-switching metrics for each of the train, validate, and test splits of the data are almost the same, indicating a similar extent of mixing in them.

5.2.1.2 Models Description

In this subsection, we describe the process of step-by-step building of our final model to perform language modeling for code-switched text that performs comparatively better than the remaining models. The path to this is approximately along the lines of incorporating the language information in the model along with the word information to improve the performance overall.

- Stage 1: With this intuition, we start with a factored LM that takes both language and words as factors to an LSTM based architecture of a language model.
- Stage 2: We then move onto improving the core of the LSTM by adapting the state-of-the-art model (Merity et al., 2018).
- Stage 3: We improve this model by incorporating lexical level language information before encoding the input (which is already done in Stage 1) and decoding the output for the next word. This is similar to a multi-task learning setting where the two tasks are predicting the next word and predicting the language of the next word. Both the losses propagate back with a weighting factor. Note that only the loss at the word decoder is used to calculate the perplexity of the model. We also perform ablations with the combinations of giving language information of the current word and predicting the language of the next word.

Factored Language Model: Each word in the vocabulary V is represented as embeddings of 2 factors: token, language information. The language information is an embedding corresponding to the language that the current token belongs to. This is described in detail later in stage 3. We build an LSTM based LM with long and short-term dependencies between switch points and partially normalized tokens.

The model trained uses two embedding layers: one learned from pure vocabulary and one learned from the language label. At training time, these embeddings are concatenated and fed into a single-layered LSTM. A linear decoding layer is then used to transition from the LSTM's representation into the vocabulary space. The model is trained using the Categorical Cross-Entropy Loss and the ADAM optimizer (Kingma and Ba, 2014). The task itself is predicting the next word given the history of word occurrences until the current time step. In addition to this, each word is anchored in the information of belongingness to a particular language.

Improving the core of the modeling architecture: The underlying model has a single-layer LSTM where the input representation from the concatenation of individual factors is fed as input. In the second stage, we improve the inner LSTM model itself based on SOTA for WikiText-2 (Merity et al., 2018). We have built a word-level language model based on Merity et al. (2017). Although the huge number of parameters in neural models enable them to learn a high degree of non-linearities present in the data, they affect the generalization capabilities of the model. In spite of techniques such as batch normalization and dropout that are generally used to regularize training in neural networks, the performance improvement has not been substantial in the case of sequential units such as recurrent models.

The nuances that this model addresses are as follows. First, coming to the problems of a default usage of dropout to the recurrent units that inhibit maintenance of longer-term dependencies. One solution for this is to use the same dropout mask at each time step, known as Locked Dropout. Another solution is the application of dropout for specific network units such as certain gates or states rather than the entire unit. The second kind of regularization techniques involve different kinds of normalization that directly impact the training process due to the induction of more trainable parameters. Finally, an effective optimization algorithm is needed

in synergy with when the dropout is applied. This paper uses ASGD (Averaged Stochastic Gradient Descent) that returns an average across the iterations that have crossed a specified loss threshold value, which is decided using a non-monotonic criterion. The distinguishing factors with this model are the regularization and optimization strategies applied to the LSTM based models built upon their core form.

Language Informed Encoding and Decoding: This is built upon the model we have from both of the above stages, where we factorize the language id of the input word and decode the next word. This factored information is decoded along with the primary task of predicting the next word. This technique is incorporated with the model described in Stage 2 with AWD-LSTM. In addition to this, we pose this as a multi-task learning problem, with shared layers except for the decoder. The two tasks are the following:

1. *Primary Task:* Predict the next word.
2. *Auxiliary Task:* Predict the language of the next word.

There are a number of ways to frame the desire for humans to switch between languages (Skiba, 1997; Moreno et al., 2002); however, we view the human desire as out of scope for this work. Instead, our focus is on how we can incorporate linguistic information while training a statistical model for code-switched text. We discuss two main choices as to where we can introduce this information: either at the input stage or at the decoding stage of an RNN language model.

N-Local Anchoring Units: Given a CS sentence $X_{cs} = (x^1, x^2 \dots, x^n)$ which has lexical level language sequence $L_{cs} = (l^1, l^2 \dots, l^n)$, our model has to predict the word at the next time step. Note that this vector l^i is the language of the *ith* lexical item and is represented as a vector of length sixteen which is trained in concert with the model. This allows our model to encode the distributional properties of the language switching. We experimented with encoding and decoding the word and language embeddings for this task. θ_{E_X} , θ_{E_L} , θ_{D_X} and θ_{D_L} are the parameters for the word encoder, language encoder, word decoder and language decoder respectively. Here the prediction of the next word is anchored N-locally in the language id information in the form of a sequence of embeddings, which is $L_{cs} = (l^1, l^2 \dots, l^n)$.

We identify four different model architectures (Figure 5.1) that could be useful in training code-switched language models. In the first model, our baseline, we have a sequence of words, and we are trying to predict the following word. This model is identical to running a traditional RNN language model on CS text. For our baseline model, we adapt the state-of-the-art language model, the AWD-LSTM, for this domain. This model is a three-layered stacked LSTM trained via Averaged SGD with tied weights between the embedding and the softmax layer. There are several other important elements of this model, all of which are detailed in Merity et al. (2017). The next word in this model is given by:

$$z = \text{Encoder}(X_{cs}, \theta_E)$$

In our second model, we extend our baseline such that we have a sequence of words and their language IDs, and we are trying to predict the following word. This model can be seen as a factored language model operating with code-switched data. So, the next word in this model is given by:

$$\text{Decoder}(\text{Encoder}(X_{cs}, \theta_{E_X}), \theta_{D_X})$$

In our third model, we take a sequence of words as input and attempt to predict both the language and the value of the following word. The next word in this model is given by:

$$\text{Decoder}(\text{Encoder}(X_{cs}, \theta_{E_X}) \oplus \text{Encoder}(L_{cs}, \theta_{E_X}), \theta_{D_X})$$

In our fourth model, we take a sequence of words and their corresponding language IDs as the input and attempt to predict both the language and value of the subsequent word. In our third and fourth models, we operate with two-loss values being calculated (one for the word error and one for the language error multiplied by 0.1), and gradients for both losses are propagated through the network and are used to update the weights.

Figure 5.2 presents a comparison of the condensed anchoring techniques for the language modeling task. Here L represents the language input data, T represents the generated output and A represents anchors in the form of language ids.

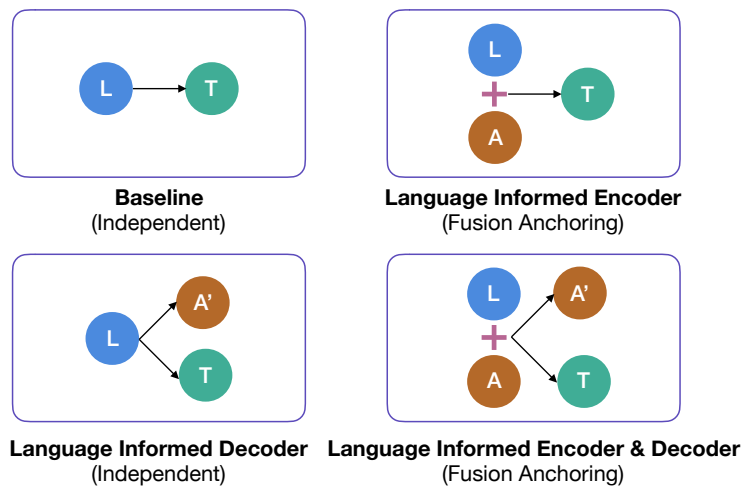


FIGURE 5.2: Comparison of the surface-form-anchoring based on lid for language modeling

5.2.1.3 Results: Quantitative Analysis

We trained 4 different models based on the description in §5.2.1.2. The results of these experiments are presented in Table 5.2. We observe that the Language Aware Encoding and Decoding with the AWD-LSTM gives the least perplexity. This aligns with our hypothesis that providing language information of the current word before encoding and enabling the model to decode the language of the next word allows the model to learn a higher-level context of switch points between the languages.

Model/Data	Train	Dev	Test
Base AWD-LSTM Model	10.08	19.73	20.92
Language Aware Encoder AWD-LSTM	10.07	19.00	20.18
Language Aware Decoder AWD-LSTM	11.60	20.72	22.01
Language Aware Encoder & Decoder AWD-LSTM	9.47	18.51	19.52

TABLE 5.2: Perplexity scores of different models from Anchoring Language Ids

अगर आप android lollipop और android phone से connected हैं gloss: if you android lollipop and android phone to connected are trans: if you are connected to android lollipop and android phone
और आप us android phone के बारे में share कर रहे हैं gloss: and you that android phone regarding share doing are trans: and you are sharing regarding that android phone
आज में अपने blog के लिए बहुत ही फायडिमेंड option लेकर आया hun जिस पर सभी high quality backlink का post लिखी gloss: today I my blog for very beneficial option bring here on which all high quality backlink 's post written trans: today I bring a very beneficial option for my blog on which all high quality backlink's post written

FIGURE 5.3: Sample Code-Mixed generation by AWD-LSTM along with gloss and translation (text in red are Hindi words that weren't identified correctly in the LID step).

5.2.1.4 Results: Qualitative Analysis:

We also did a qualitative analysis of this model with the sentences they generate. Some generation examples of the baseline AWD-LSTM are shown in table 5.3. We observe them to be quite consistent in generating meaningful code-mixed n-grams of length ten and above.

However, the sentences generated were not very coherent. Part of the reason for this is training sentences themselves did not have proper punctuation and end-markers, which are issues to take care of during pre-processing. Notice that in the first two examples, the first word is connective (in specific a conjunction), and hence expects their respective sub-ordinate and co-ordinate parts of the missing sentences.

Also, as seen in the second word of the last sentence, which is the transliterated form of the word for 'mei' in Hindi (which is the Hindi word for 'in'), in this context, the word that is closest in lexical form should have been 'main' (which is the Hindi word for 'I'). Using transliteration as a proxy to standardize the representations of the words is the reason for the model to make such errors. Character level models have been working well for spell corrections and normalization, and incorporating character level convolution as input along with the word embeddings could potentially help solve this problem.

Simultaneously, we observe the t-sne plots of the Hindi and English words from the embedding layer separately. For the English terms, the content words like *infographics*, *click*, *cyber*, *blueborne* are related to blogs and tend to be grouped closer. While in case of the Hindi terms, verbs like *banaya*(-made), *aaoge*(-come), *karte hain*(are doing) and function words are grouped closer. This is understandable because of the matrix language being in Hindi. So, the Hindi words made up the bulk of the syntax contained in these sentences, with English content words sprinkled in between. Also, because a long distance separates the English content words, they do not influence the meanings of one another in the context. For the same reason we did not find any noticeable correlation between the meanings of Hindi and English word embeddings because they played very different roles in sentences.

5.2.2 N-Local Anchoring for Speech Synthesis:

In §5.2.1, we have seen how explicit modeling of lexical level language ids aids in code-switched language modeling. In this subsection, we are going to delve into the utility of these explicit N-Local anchors in improving the quality of generating mixed-language instructions. Specifically, we will look into the domain of navigational instructions in settings with a mismatch between the language in which a text-to-speech (TTS) system is trained and the language from which the local place names are derived.

Text-to-Speech (TTS) systems that can read navigation instructions are one of the most widely used speech interfaces today. Text in the navigation domain may contain named entities such as location names that are not in the language that the TTS database is recorded in. Moreover, named entities can be compound words where individual lexical items belong to different languages. These named entities may be transliterated into the script that the TTS system is trained on. This may result in incorrect pronunciation rules being used for such words. We describe experiments to extend our previous work in generating code-mixed speech to synthesize navigation instructions with a mixed-lingual TTS system. We conduct subjective listening tests with two sets of users, one being students who are native speakers of an Indian language and very proficient in English, and the other being drivers with low English literacy but familiarity with location names. We find that in both sets of users, there is a significant preference for our proposed system over a baseline system that synthesizes instructions in English.

Navigation systems that can render instructions in the form of synthesized speech in addition to a visual interface are an important application of TTS Systems where being hands-free is critical. The text that needs to be synthesized in the navigation domain contains many named entities, such as names of roads and landmarks. The language that the TTS system is trained on may not be the same as the language that local place names are derived from. This may lead to pronunciation that does not seem natural, which may affect the usability of such systems. Text for instructions is typically rendered in a single script. That is, although names of roads and landmarks are derived from a particular language, they are represented in the language that the TTS system is speaking in. For example, instructions being spoken by an Indian English TTS system for navigation in Bangalore will contain location names transliterated into the Roman script. Language identification can be applied to categorize words in text that contains foreign named entities so that corresponding phonetic rules are applied to each set accordingly. This scenario is different from code-mixing in the sense that only certain words, specifically proper nouns, belong to the native language. However, the influence of English still prevails in the names of the places as well, for example: ‘road’, ‘park’, ‘mall’, ‘plaza’ etc. An example navigation instruction that is collected between two locations in Delhi is:

Turn\Eng left\Eng at\Eng Mukhiya\Hin Market\Eng Chowk\Hin onto\Eng Karawal\Hin Nagar\Hin.

In this example, the words followed by ‘\Eng’ and ‘\Hin’ are English and Hindi words, respectively. As stated before, there is a mixture of languages in the names of the places as well. For example, in ‘Mukhiya Market Chowk’, ‘market’ is an English word while the others are derived from the native language Hindi. In this work, we extend the work on synthesizing

code-mixed text using a monolingual voice to the domain of synthesizing navigation instructions using a bilingual voice. We build systems to synthesize navigation instructions using a Hindi-English bilingual voice for location names derived from Hindi, Kannada, and Telugu. In addition, we conduct subjective listening tests to compare our system with a monolingual baseline system. Studies show that the performance of a driver is impacted by their cognitive load (Jonides, 1981). This may compromise the ability of the driver to perceive safety-critical events. Considering that TTS systems for navigation instructions are deployed in real-time, it is imperative to aid the user with the provision of more natural auditory instructions. With the proliferation of ride-sharing applications like Uber and Ola in countries like India, many individuals working as full-time drivers are now using navigation apps that have TTS systems. In some cases, these drivers choose to use such apps voluntarily, while in other cases, the use of such apps is mandated by the cab company. Many of these drivers are semi-literate and have low English proficiency, and we conduct interviews and listening tests with them to evaluate our system. We also conduct listening tests with another set of users, mostly comprising of graduate students who have high English proficiency. In the remainder of the paper, we refer to an English navigation instruction as native to a language if it has words derived from that language as the names of the places.

5.2.2.1 Data Collection and Description

We used the Google Maps API to collect navigation directions from the locations where the following are the native languages: Hindi, Telugu, Kannada, Gujarati, Bengali, Marathi, and Tamil. While we conducted listening tests for Hindi, Kannada, and Telugu, this method is easily extensible to the other languages as well. The choice of these languages was based on access to native speakers in these languages to perform subjective testing. The navigation instructions used in GPS applications are in English, and so the syntactic structure of these instructions remains in English. The names of the places, including native language words, are considered words from the *embedded language* into English, which is the *matrix language*, in the matrix language-embedded language theory of code-mixing. Language Mix Ratio (LMR) is defined as the ratio of the number of words from the embedded language to the number of words in the matrix language. Table 5.3 includes details about the data, including the LMR, after using the language identification module mentioned in the following section.

Language	# distinct routes	# sentences	LMR
Hindi	399	4,806	0.2392
Telugu	1,974	19,976	0.1576
Kannada	8,898	108,178	0.1471
Gujarati	1,995	17,649	0.0942
Bengali	2,448	24,909	0.1852
Marathi	2,363	23,614	0.1977
Tamil	3,322	37,428	0.1612

TABLE 5.3: Navigation Instructions: Data Statistics

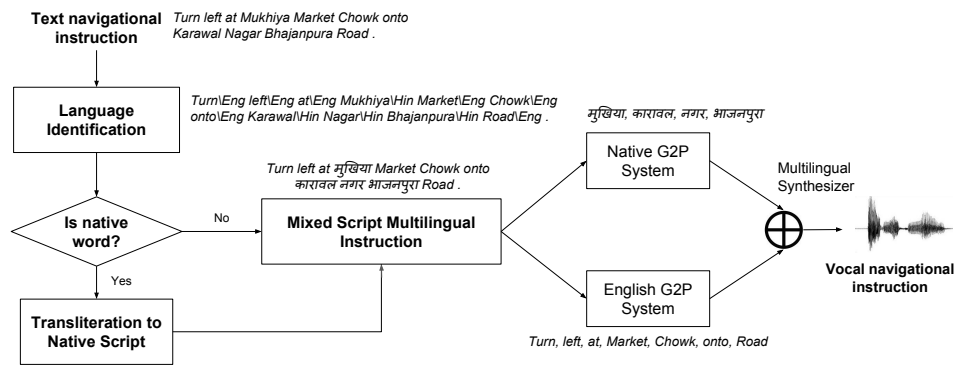


FIGURE 5.6: Architecture of the system with example of Hindi navigation instruction (Note that the language of the word ‘Chowk’ is misidentified and transliteration of ‘karawal’ is incorrect)

The navigation domain has fewer spelling variations than general cross script code-mixing in social media and blogging websites observed from our previous work in language modeling, where normalization is crucial. The navigation data we collected has fairly standardized spellings for the names of the places, although the native words of the places are transliterated into English.

5.2.2.2 Model Description

Our proposed technique is similar to the pipeline we follow for synthesizing code-mixed text - first, we identify the language of each individual word in the sentence. These language ids are the N-Local anchor units that aid the synthesis process. Then, we transliterate the words that are not in English to the native script. This mixed script multilingual instruction is sent to corresponding G2P systems based on the language of lexical items. Finally, a multilingual synthesizer is used to generate vocal navigational instruction. This section briefly outlines these stages. The overall architecture of the system is demonstrated in Figure 5.6.

Anchor Extraction - Language Identification: In this stage, the task is to identify the names of places in the native language in each of the navigation instructions. One way is to use POS taggers and Named Entity Recognition tools to identify the names of locations in the instructions. We have attempted mapping named entities from Wikipedia full text dumps with the ones found in navigational instructions by using Soundex encodings. This method has good coverage of important places but did not work well for local street names. In addition, as discussed in the introduction, we often find that place-names contain English words like ‘mall’, ‘park’, ‘station’ etc., which need to be pronounced with English pronunciation rules. Hence, we identify the language of each word in the navigation instructions. We used an off-the-shelf system for language identification (Bhat et al., 2014) which uses character n-grams as features. Due to the specificity of the domain, we also attempt to mitigate errors made by the system by labeling common words like ‘road’, ‘bus’, ‘main’ as English words. This system covers all the languages of our interest, except for Marathi. Since this system is not trained to identify

Marathi and English, we proxy Hindi for Marathi for the language identification task. Though this is not ideal, it serves as a solution to differentiate English and non-English words.

Transliteration: To map the representation of native words to their corresponding phonemes used in the front end, these words are transliterated from the Romanized script to the native script. [Bhat et al. \(2014\)](#) modeled transliteration as a structured prediction problem using second order Hidden Markov Models. In our initial experiments using Soundex codes, we mapped these transliterated words to words from large text of monolingual script (including wiki text dump, wiki titles, and web pages from relevant queries) to derive a locality name. Even a very large amount of text had coverage issues with respect to proper nouns. We experimented with transliteration as a sequence to sequence problem by training an LSTM to convert English sequences for the names of the places to the native script. We used 1000 parallel examples of Hindi words written in Devanagari and Romanized scripts from the FIRE task data ([Choudhury et al., 2014](#)) for this task. When trained on 800 samples and tested on 200 samples, the character level accuracy is 35.64%, while the word-level accuracy is much smaller. The problems of recurring and invalid sequences of characters were addressed by building a language model of the native script. In similar lines of ([Black et al., 1998](#)), which uses decision tree based letter to sound rules, we adapted this approach for the task of transliteration, and for the same test set, we got a word-level accuracy of 26%.

Brahmi-Net transliteration ([Kunchukuttan et al., 2015](#)) considers this problem similar to a phrase-based translation problem, through which sequences of characters from source to the target language are learnt, where the parallel corpus is trained using Moses. This system supports 13 Indo-Aryan languages, 4 Dravidian languages, and English, including 306 language pairs for statistical transliteration. Using this, the accuracy corresponding to the correctness of the entire word for the 200 test examples is 32.65%. Since this yielded higher accuracies at the word level, we proceeded with this scheme using their REST API to transliterate words into their native script.

Synthesis: The final step is to synthesize the navigation instructions that are transliterated into the appropriate script. Once we transliterate native language words, we synthesize the sentence using the bilingual TTS voice.

Speech data from Mono and English sets of the male speaker released as a part of resources for Indian languages ([Baby et al., 2016](#)) was used for these experiments. We used all the 1,132 prompts from the Arctic set recorded by a male Indian English speaker and used only the first 600 prompts from the Hindi set so that both Hindi and English utterances are of equal duration (approximately an hour each), as the Hindi utterances were longer. The speech data was sampled at 16 kHz and recorded by a professional speaker in a high-quality studio environment. For combining the English and Hindi phonesets, we used a simple phone clustering approach: the phones common in English and Hindi were retained as is, and the phones present only in English were added, resulting in a common phoneset. By doing this, we bypassed the phone-mapping process, which was shown to result in accented speech ([Elluru et al., 2013](#)) and would have limited the phones that could be used to those in the target language's phoneset. For getting pronunciations of native language words, we used the Festvox Indic frontend ([Parlikar](#)

et al., 2016), which provides a g2p mapping between all Indian language UTF-8 code points and a phoneme from a common Indic phoneset. For some languages, rules like stress assignment, schwa deletion, and voicing rules are implemented in the frontend. To build the voice, we followed the standard CLUSTERGEN (Black, 2006) Statistical Parametric Synthesis voice building process.

Figure 5.7 presents a comparison of the condensed forms of the models with anchors. The generated output is in the form of speech which is represented as S here.

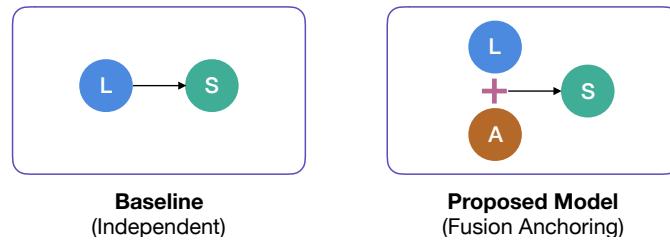


FIGURE 5.7: Comparison of the surface-form-anchoring based on lid for speech synthesis

5.2.2.3 Results

To perform preference testing, we synthesized navigation instructions using two methods. The first method was to retain all the lexical items in English. The second method used the proposed technique, i.e., language identification, transliteration, and g2p using the native script. Both the methods used the same TTS voice trained using the bilingual data. We will now present our findings from preference testing and user studies with drivers.

Preference Testing: We conducted a user preference study to compare the baseline system to our proposed approach. We randomly sampled 20 navigation instructions in each of Hindi, Kannada, and Telugu languages from the data collected and synthesized them. We used the Testvox web-based framework (Parlikar, 2012) for conducting these listening tests. Examples of these synthesized files can be found here ². We asked five native speakers, each of Hindi, Telugu, and Kannada, to perform the listening test. We gave each speaker navigation instructions with location names derived from their mother tongue. We asked them to pick the sample that sounded more natural and understandable, with an option of choosing ‘No preference’ as well. Table 5.4 presents the results of this preference testing for three languages; Hindi, Kannada, and Telugu. We can see that there was a significant preference for our proposed system in all three languages.

In addition to preference testing, we also did an informal study for intelligibility. For each of the languages, one student was asked to transcribe 20 navigation instructions, and we recorded the number of times that the person had to listen to it to transcribe the sentence accurately. On an average, the transcriber had to listen 1.70 times for Hindi, 1.75 for Telugu, and 2.15 for Kannada navigation instructions.

²<http://www.cs.cmu.edu/~kchandu/navigation/index.html>

Language	Prefer Baseline	Prefer Proposed	No Preference
Hindi	17%	70%	13%
Telugu	4%	76%	20%
Kannada	19%	69%	12%

TABLE 5.4: Subjective listening tests for preference in synthesis of mixed-language navigational instructions

The language identification module that we are using has an accuracy of 88.08%, 92.27%, and 91.89% for Hindi-English, Telugu-English, and Kannada-English language pairs. Some words are very ambiguous, and the limited context may not be enough to identify the language correctly, particularly if the language identification system is trained on data from another domain. For example, the word ‘to’ is identified as a Hindi word as it is very common in Hindi (meaning: ‘then’); however, in the navigation instructions, it is always an English word. We observed the following errors in the Kannada native words. The language identification system, apart from using n-gram character features, also takes into account the context information from surrounding words. Hence the same word can be identified in different languages based on context. One such example is ‘Jaraganahalli’, identified as Kannada and English in two different instructions. Erroneous transliteration introduces some errors, for example, for words like ‘Hosakerehalli’ and ‘Gubbi Thotadappa Road’. People acquainted with these locations, however, could still recognize them. As observed from Table 5.4, our system is preferred to a great extent in Telugu in comparison to other languages that we conducted this study. This could be because Telugu words are relatively longer than in the other languages, and hence English pronunciations of long Telugu words may be even more distracting.

User study with drivers: In addition to conducting listening tests with users with high English proficiency and familiarity with speech-based systems, we also wanted to conduct user studies with a population of drivers who use navigation apps. These drivers are typically semi-literate and have low English proficiency and relatively low exposure to technology.

We conducted interviews and listening tests with 11 subjects who are full-time drivers in Bangalore. We briefed the drivers about the goals of the project, collected demographic data from them, and asked them about their experience with GPS-based navigation systems, particularly about the TTS part of the systems. The drivers were given a mobile top-up recharge of INR 50 (around 0.8 USD) as compensation for participating in the study. The entire interview was conducted in Kannada, the local language in Bangalore, although the TTS system itself was the bilingual voice described above. All the drivers in the study reported that they were familiar with locations in Bangalore, and almost all of them had lived in Bangalore for at least five years. Most drivers said that they had low English proficiency, with almost all of them saying that they could not speak or write English, but they could read and understand some English. All the drivers were multilingual, with some drivers knowing as many as five languages - Kannada, Tamil, Telugu, and Hindi being the most common languages that drivers knew, with some drivers knowing some English and one driver also knowing Urdu. After the initial interview to collect demographic information, the drivers were given the same listening

task as the previous study, with location names in Bangalore. Each driver listened to ten pairs of audio files using the Testvox interface. They were asked to choose the system that they could understand better, and one of the authors helped them navigate the web-based listening test and answered any questions they had. Table 5.5 shows their listening preference between the baseline system and our proposed approach.

Prefer Baseline	Prefer Proposed	No Preference
34%	60%	6%

TABLE 5.5: Subjective listening tests with drivers for synthesis of mixed language navigational instructions

From Table 5.5, it is clear that drivers had a strong preference for the proposed system. In many cases, they also pointed out specific words that they could understand better in the proposed system. The proposed system produced some extra schwas in some words, which made it sound slightly unnatural, but the drivers did not point this out. In some cases, the drivers also pointed out that the (incorrect) pronunciation of a particular word in the monolingual system was similar to what they heard in the current navigation app that they used.

After the listening test, we asked drivers open-ended questions about their experience with navigation apps and suggestions for improvement. Some drivers had driven Ola and Uber cabs and had more experience with navigation apps, while others used them only when they went out of town, did not know a route, or wanted to find out about traffic conditions. Surprisingly, almost all drivers preferred the navigation instructions to be in English rather than the local language or their native language. Their reasons for this were that the instructions used minimal English, which they already understood, and they wanted the instructions to be in a language that their passengers could understand so that there was more transparency with the customer. They did, however, say that they knew of other drivers who knew no English who used the navigation app with the voice on mute because they could not understand it.

In this section, we examined the role of N-Local anchoring units in the form of language ids to explicitly model each lexical unit in the tasks of language modeling and speech synthesis. However, it is often the case where it is hard to derive these language id labels, especially because code-switching in written forms involves cross-scripting. For instance, in the case of Hinglish, we often observe Hindi words to be written in the Roman script instead of the Devanagari script. Annotating lexical items with language information is manually intensive work while automating this process is still an active area of research. Hence we pave the way unto our next section, which anchors the generation of code-switched text at the entire sentence level instead of each of the lexical units.

5.3 N-Local Anchoring in Personality

In the previous sections, we have explored the choices in surface form representation in multi-lingual scenarios, especially when the participating languages interact with one another within

the same sentence. In this section, we are going to explore surface realization in monolingual scenarios, realized in the form of personalities through visual storytelling. This is ongoing research work, and hence this section briefly describes techniques that we adapted for personality-induced story generation, which motivates the proposed research work in this direction.

As we have discussed previously, visual storytelling is the task of generating stories based on a sequence of images. Inspired by the recent works in neural generation focusing on controlling the *form* of text, this chapter explores the idea of generating these stories in different personas. However, one of the main challenges of performing this task is the lack of a dataset of visual stories in different personas. Having said that, there are independent datasets for both visual storytelling and annotated sentences for various persona. This section describes an approach to overcome this by getting labeled persona data from a different task and leveraging those annotations to perform persona-based story generation. This section presents an inspection of various ways of incorporating personality in both the encoder and the decoder representations to steer the generation in the target direction by anchoring in the persona representation. To this end, five models are proposed, which are incremental extensions to the baseline model to perform the task at hand. In our experiments, five different personas are used to guide the generation process. The models based on our hypotheses perform better at capturing words while generating stories in the target persona.

This section introduces an approach to generating visual stories in five different personality types. A key challenge to this end is the lack of large-scale persona annotated stories. This is addressed by transferring knowledge from annotated data in the dialog domain to the storytelling domain. The visual story generator model is based on [Kim et al. \(2018\)](#), and I propose multiple techniques to induce the personalities in the latent representations of both the encoder and the decoder. Our work aims to learn the mapping between the latent representations of the images and the tokens of the story such that the generative model is encouraged to generate tokens of a particular personality. The generative models are evaluated using the automatic metric of ROUGE ([Lin, 2004](#)) which takes into account the sentence level similarity in structure and thus roughly evaluates the matching of content. We acknowledge that there is a drop in this metric since our model is not trying to optimize generation alone but also adapts personality from a different dataset.

The success of generating the story in the target personality type is also evaluated using automatic and qualitative analysis. The automatic metrics comprise the classification accuracies rooted in the annotated data. It is observed that one of the proposed models (LEPC, described in [§5.3.2](#)) performs slightly better at classification accuracies for most of the personas while retaining similar ROUGE scores.

The main contribution of this section is showing simple yet effective approaches to narrative visual stories in different personality types. This section also displays an effective way of using annotated data in the dialog domain to guide the generative models to a specified target personality.

5.3.1 Datasets Description

Coalescing the segments of personality and sequential generation together, our task is to generate a grounded sequential story from the view of a personality. To bring this to action, we describe the two sources of data we use to generate personality-based stories in this section. The first source of data is focused on generic story generation from a sequence of images and the second source of data includes annotations for personality types for sentences. We tailor a composition of these two sources to obtain a dataset for personality-based visual storytelling. Here, we note that the techniques described above can be applied for unimodal story generation as well.

Visual Story Telling: Visual Storytelling is the task of generating stories from a sequence of images. A dataset for this grounded sequential generation problem was collected by Huang et al. (2016), and an effort for a shared task³ was led in 2018. The dataset includes 40,155 training sequences of stories. It comprises a sequence of images, descriptions of images in isolation, and stories of images in sequences. We randomly divide the dataset into five segments (comprising of 8031 stories each), and each segment is associated with a personality.

Personality Dialog: Shuster et al. (2018) have provided a dataset of 401k dialog utterances, each of which belongs to one of 215 different personalities. The dataset was collected through image-grounded human-human conversations. Humans were asked to play the role of a given personality. This makes this dataset very pertinent for our task as it was collected through engaging image chat between two humans enacting their personalities.

For our task, we wanted to choose a set of five distinct personality types. Let the set of utterances that belong to each personality type be $U_p = \{u_p^1, \dots, u_p^n\}$ where $p \in \{1, \dots, 215\}$. We first calculate the pooled BERT representation (Devlin et al., 2018) of each of the utterances. To get the representation of the personality \mathcal{P} , we simply average the BERT representations of all the utterances that belong to that personality. The representation of each personality is given by:

$$\mathcal{P}_p = \frac{\sum_{k=1}^n \text{BERT}(u_p^k)}{n} \quad (5.1)$$

This representation is calculated only on the train set of Shuster et al. (2018).

Since our goal is to pick the five most distinct personality types, we have the daunting task of filtering the 215 personality types to 5. To make our task easier, we want to group similar personalities together. Hence, we use K-Means Clustering to cluster the representations of the personalities into 40 clusters⁴. We get well-formed and meaningful clusters which look like [Impersonal, Aloof (Detached, Distant), Apathetic (Uncaring, Disinterested), Blunt, Cold, Stiff]; [Practical, Rational, Realistic, Businesslike]; [Empathetic, Sympathetic, Emotional]; [Calm, Gentle, Peaceful, Relaxed, Mellow (Soothing, Sweet)], etc., We then build a classifier using

³<http://visionandlanguage.net/workshop2018/index.html#challenge>

⁴We do not perform an exhaustive search on the number of clusters. We tried k values of 5, 20, and 40 and selected 40 as the ideal value based on manual inspection of the clusters.

the technique described in §5.3.2 to classify the utterances to belong to one of the 40 clusters. We pick the top five clusters that give the highest accuracy for the 40-way classification.

The five personality clusters selected are:

- Cluster 1 (C1): Arrogant, Conceited, Egocentric, Lazy, Money-minded, Narcissistic, Pompous and Resentful
- Cluster 2 (C2): Skeptical and Paranoid
- Cluster 3 (C3): Energetic, Enthusiastic, Exciting, Happy, Vivacious, Excitable
- Cluster 4 (C4): Bland and Uncreative
- Cluster 5 (C5): Patriotic

5.3.2 Model Description

As mentioned in earlier chapters on the properties of content, we have a dataset of visual stories $S = \{S_1, \dots, S_n\}$. Each story S_i is a set of sequence of five images and the corresponding text of the story $S_i = \{(I_i^{(1)}, x_i^{(1)}), \dots, (I_i^{(5)}, x_i^{(5)})\}$. Our task is to generate the story based on not only the sequence of the images but also closely following the narrative style of a personality type. We have five personality types (described in §5.3.1) $P = \{p_1, \dots, p_5\}$ and each story is assigned one of these five personalities as their target persona. Here, each p_i represents the one-hot encoding of the target personality for story i.e $p_1 = [1, 0, 0, 0, 0]$ and so on till $p_5 = [0, 0, 0, 0, 1]$. Hence, we create a dataset such that for each story, we also have a specified target personality type $S_i = \{(I_i^{(1)}, x_i^{(1)}), \dots, (I_i^{(5)}, x_i^{(5)}); p_i\}$. The inputs to our models are the sequence of images and the target personality type. We build generative models such that they are able to generate stories in the specified target personality type from the images. In this section, we first briefly describe classifiers trained discriminatively to identify each of the personalities and then move on to the story generation models that use these classifiers.

Here is an overview of the differences in the six models that we describe next.

1. The baseline model (Glocal) is a sequence to sequence model with global and local contexts for generating story sentences corresponding to each image.
2. The Multitask Personality Prediction (MPP) model is equipped with predicting the personality in addition to generating the sentences of the story. This model also incorporates binary encoding of personality.
3. The Latent Encoding of Personality in Context (LEPC) model incorporates an embedding of the personality as opposed to binary encoding.
4. The Latent Encoding of Personality in Decoder (LEPD) model augments personality embedding at each step in the decoder, where each step generates a token.

5. Stripped Encoding of Personality in Context (SEPC) is similar to LEPC but encodes personality embedding after stripping the mean of the story representation.
6. Stripped Encoding of Personality in Decoder (SEPD) is similar to LEPC but encodes personality embedding after stripping the mean of the story representation. This is similar to the intuition behind SEPC.

Classification We use convolutional neural network (CNN) architecture to train our classifiers. We train five separate binary classifiers for each of the personality types. The classifiers are trained to predict whether a sentence belongs to a particular personality or not. We train the classifiers in a supervised manner. We need labeled data to train each of the classifiers. Each sample of text x in the respective datasets of each of the five personality types has a label in the set $\{0, 1\}$. Let $\theta_C^{p_j}$ denote the parameters of the classifier for personality p_j where $j \in \{1, \dots, 5\}$. Each classifier is trained with the following objective:

$$\mathcal{L}(\theta_C^{p_j}) = \mathbb{E}_x[\log q_C(p_j|x)] \quad (5.2)$$

We use cross entropy loss to calculate $\mathcal{L}_C^{p_j}$ for each of the five classifiers. The classifiers accept continuous representations of tokens as input.

Story Generation We present five extensions to incorporate personality-based features in the generation of stories.

(1) Baseline model (Glocal): We first describe the baseline model that is used for visual storytelling. This is based on the model Kim et al. (2018) that attained better scores on human evaluation metrics. It follows an encoder-decoder framework translating a sequence of images into a story. From here on, we refer to this model as *glocal* through the rest of the section owing to the global and local features in the generation of story sequence at each step (described in this section).

The image features for each of the steps are extracted with a ResNet-152 (He et al., 2016b) post resizing to 224 X 224. The features are taken from the penultimate layer of this pretrained model and the gradients are not propagated through this layer during optimization. These features are passed through a fully connected layer to obtain the final image features. In order to obtain an overall context of the story, the sequence of the image features is passed through a Bi-LSTM. This represents the global context of the story. For each step in the generation of the story, the local context corresponding to the specificity of that particular image is obtained by augmenting the image features (local context) to the context features from the Bi-LSTM (global context). These *glocal features* are used to decode the story sentence at each step. This concludes the encoder part of the story. The decoder of each step in the story also uses an LSTM, which takes the same *glocal feature* for that particular step at each time step. Hence there are five *glocal features* feeding into each time step in the decoder.

For simplicity in understanding, we use the following notations throughout model descriptions to represent the mathematical formulation of the generation models. Subscript k indicates the

k^{th} step or sentence in a story. Subscript i indicates the i^{th} story example. The story encoder is represented as *Encoder* which comprises the features extracted from the penultimate layer of ResNet-152 concatenated with the global context features from the Bi-LSTM. The entirety of this representation in encoder and the global features obtained is represented using z_k for the k_{th} step or sentence in the story.

$$z_k = Encoder(I_k) \quad (5.3)$$

Now, the generation of a sentence in the story is represented as follows:

$$\hat{x}_k \sim \prod_t Pr(\hat{x}_k^t | \hat{x}_k^{<t}, z_k) \quad (5.4)$$

The generated sentence \hat{x}_k is obtained from each of the output words \hat{x}_k^t which is generated by conditioning on all of the prior words $\hat{x}_k^{<t}$ and the global feature obtained as z_k .

Personality based Generation: In the rest of the section, we are going to describe the incremental extensions to the baseline to adapt the model to perform persona-based story generation.

(2) Multitask Personality Prediction (MPP): The intuition behind the hypothesis here is to provide the personality information to the model and enable it to predict the personality and the generation of the story. The obvious extension to provide personality information is to incorporate the one-hot encoding $p_i \in P$ of the five personas in the context before the decoder. The visual storytelling data is split into five predetermined personalities as described in §5.3.1. For each story, the corresponding personality is encoded in a one-hot representation and is augmented to the global context features. These features are then given to the decoder to produce each step in the story. The model is enabled to perform two tasks: the primary task is to generate the story, and the secondary task is to predict the personality of the story. The classifiers described in §5.3.2 are used to perform personality prediction. Formally, the generation process is represented by:

$$\hat{x}_k \sim \prod_t Pr(\hat{x}_k^t | \hat{x}_k^{<t}, z_k, p_i) \quad (5.5)$$

Here, we condition the generation of each word on the global context features z_k , binary encoding of the personality p_i and the words generated till that point.

The cross entropy loss for generation is \mathcal{L}_g and the loss for the prediction of each of the personalities is $L_C^{P_j}$ given by Eq 5.2. The overall loss optimized for this model is:

$$\mathcal{L}_{total} = \alpha \cdot \mathcal{L}_g + \frac{(1 - \alpha)}{5} \cdot \sum_{j=1}^5 \mathcal{L}_C^{p_j}$$

The overall model is optimized on this total loss. We use cross entropy loss for each of the individual losses. We give a higher weight α to the story generation and equally distribute the remaining $(1 - \alpha)$ among each of the 5 personalities.

(3) Latent Encoding of Personality in Context (LEPC): This model is an incremental improvement over the MPP model. The key difference is the incorporation of personality as an embedding that captures more centralized traits in words belonging to that particular personality. For each of the five personality types, we have a latent representation of the personality (\mathcal{P}), as opposed to the binary encoding in MPP model. Similar to the earlier setting, this average personality feature vector is concatenated with the global context vector. The generation step is formally represented as:

$$\hat{x}_k \sim \prod_t Pr(\hat{x}_k^t | \hat{x}_k^{<t}, [z_k; \mathcal{P}], p_i) \quad (5.6)$$

This means that z_k is concatenated with \mathcal{P} to give personality informed representation; and the generation of each word is conditioned on these concatenated features z_k , binary encoding of the personality p_i and the words generated so far.

(4) Latent Encoding of Personality in Decoder (LEPD): Instead of augmenting the personality traits to the context as done in the LEPC model, they could be explicitly used in each step of decoding. The latent representation of the personality (\mathcal{P}) is concatenated with the word embedding for each time step in the decoder.

$$\hat{x}_k \sim \prod_t Pr(\hat{x}_k^t | [\hat{x}_k^{<t}; \mathcal{P}], z_k, p_i) \quad (5.7)$$

The generation of each word is conditioned on the words generated so far that are already concatenated with the average vector for the corresponding personality, the global features, and the binary encoding of the personality.

(5) Stripped Encoding of Personality in Context (SEPC): In order to orient the generation more towards the personality, we need to go beyond the simple augmentation of personality. Deriving motivation from neural storytelling⁵, we use a similar approach to subtract central characteristics of words in a story and add the characteristics of the personality. Along the same lines of calculating an average representation for each of the personalities, we also obtain an average representation of the story \mathcal{S} . This average representation \mathcal{S} intuitively

⁵<https://github.com/ryankiros/neural-storyteller>

captures the style of the story. Essentially, the story style is being stripped off the context, and personality style is incorporated. The modified glocal feature that is given to the decoder is obtained as $m = z_k - \mathcal{S} + \mathcal{P}$. The generation process is now conditioned on m instead of z_k . Hence, the generation of each word in decoding is conditioned on the words generated so far ($\hat{x}_k^{<t}$), the binary encoding of the personality (p_i) and the modified representation of the context features (m).

$$\hat{x}_k \sim \prod_t Pr(\hat{x}_k^t | \hat{x}_k^{<t}, m, p_i) \quad (5.8)$$

Here, note that the context features obtained thus far are from the visual data and performing this operation is attempting to associate the visual data with the central textual representations of the personalities and the stories.

(6) Stripped Encoding of Personality in Decoder (SEPD): This model is similar to SEPC with the modification of performing the stripping at each word embedding in the decoder as opposed to the context level stripping. The time steps to strip features are at the sentence level in SEPC and are at word level in the SEPD model. The LSTM based decoder decodes one word at a time. At each of these time steps, the word embedding feature \mathcal{E} is modified as $e_k = \mathcal{E} - \mathcal{S} + \mathcal{P}$.

This modification is performed in each step of the decoding process. These modified features are used to generate each sentence in the full story. The model is trained to generate a sentence in the story as described below:

$$\hat{x}_k \sim \prod_t Pr(\hat{x}_k^t | e_k^{<t}, z_k, p_i) \quad (5.9)$$

The generation of each word is conditioned on the modified word embeddings using the aforementioned transformation ($e_k^{<t}$), the binary encodings of the personalities (p_i), and the glocal context features.

The various models discussed above are presented in the condensed anchor forms in the Figure 5.8.

5.3.3 Experiments and Results

This section presents the experimental setup for the models described in §5.3.2. Each of the models is an incremental extension over the baseline glocal model. The hyperparameters used for this are as follows.

We build five separate classifiers, one for each personality cluster. Note that these clusters are also associated with personalities and hence are later referred to as P followed by the cluster id in the following sections. To build the five binary classifiers, we create label balanced datasets for each cluster i.e., we randomly select as many negative samples from the remaining 4 clusters

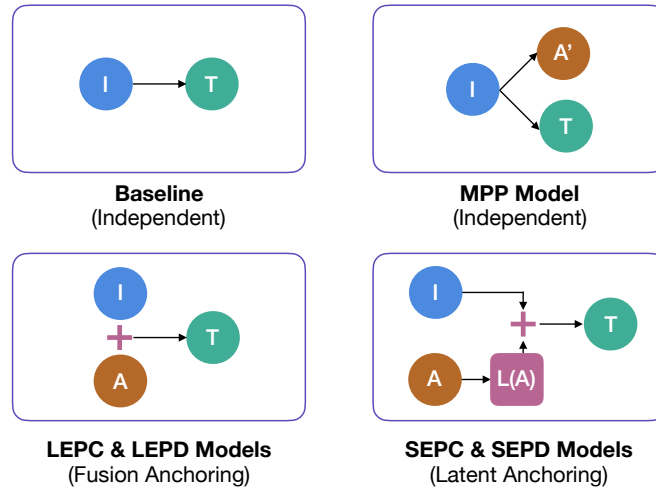


FIGURE 5.8: Comparison of the surface-form-anchoring based on multimodal interactions for persona based generation

as there are positive samples in that cluster. We use the train, dev and test split as is from Shuster et al. (2018). The dataset statistics for each of the five clusters is provided in Table 5.6.

Cluster Type	Train	Dev	Test
Cluster 1	26538	1132	2294
Cluster 2	6614	266	608
Cluster 3	19784	898	1646
Cluster 4	6646	266	576
Cluster 5	3262	138	314

TABLE 5.6: Statistics of data belonging to each of the persona clusters

Note that all the datasets have a balanced distribution of labels **0** and **1**. For our experiments, it does not matter that the distribution of the number of samples is different because we build separate classifiers for each of the clusters, and their output is treated as independent from one another.

As seen in Table 5.7, all the classifiers attain good accuracies and F-scores on the test set.

	C1	C2	C3	C4	C5
Acc.	79.12	81.09	83.17	77.95	84.08
F1	0.79	0.81	0.83	0.78	0.84

TABLE 5.7: Performance of classifiers for each of the persona clusters

We finally calculate the representation \mathcal{P} for each of the five clusters and the representation \mathcal{S} of stories using equation 5.1. Note that \mathcal{S} is calculated over the visual storytelling dataset. These representations are used by our generative models **LEPC**, **LEPD**, **SEPC**, and **SEPD**.

Hyperparameters: The hidden size of the Bi-LSTM encoder of the story to capture context is 1024. The dimensionality of the global context vector z_k is 2048. A dropout layer of 50% is

Original	grandma loves when all the kids come over to visit .	she will pick them them up and put them on her lap even though it <unk> .	the kids love each other as well giving lots of hugs and love .	grandma can not forget her little girl and gives her some love as well .	grandpa says it 's time for cake .
Glocal	the family is having a great time .	they are playing with each other .	he is happy to see his grandson .	she is being silly	the birthday girl is eating a cake .
MPP	[male] and his friends are having a great time .	they are all smiles for the camera .	everyone is enjoying their new family .	[female] is so excited to be there .	she is very happy about her birthday .
LEPC	the family was having a great time .	they were so happy to be together .	they were having a good time with grandson .	she was very excited to play with a kid .	he was surprised by all of his friends .
LEPD	the family was ready to see a lot of a party .	they had a great time .	they were having a lot of fun .	we had a great day .	he was happy to eat cake .
SEPC	the parade was very beautiful .	there were a lot of people there .	we were so happy to be a great time .	i had a great time .	this was a picture of a little girl .
SEPD	the family is a great time .	it was a lot of a big .	there were a lot .	i had a picture .	they were a very .

FIGURE 5.9: Comparison of generated *stories* from all the described models.

applied post the fully connected layer to obtain the image features and after the global features obtained from Bi-LSTM, which is 2-layered. The word embedding dimension used is 256. The learning rate is 1e-3 with a weight decay of 1e-5. Adam optimizer is used with batch normalization and a momentum of 0.01. Weighting the loss functions differently is done to penalize the model more if the decoding is at fault compared to not predicting the story's personality. α is set to 0.5, and each of the individual personality losses is weighted by a factor of 0.1.

The rest of the 5 models use the same hyperparameter setting with an exception to word embedding dimension. The average personality (\mathcal{P}) and the average story (\mathcal{S}) representations are obtained from pre-trained BERT model. Hence this is a 768 dimensional vector. In order to perform the stripping of the story feature and adding the personality features to the word embeddings in the decoder, the word embedding dimension is matched to 768 in the SEPD model.

Model	C1	C2	C3	C4	C5
Glocal	69.90	73.29	51.55	34.91	65.86
MPP	69.35	72.44	47.54	33.83	58.49
LEPC	70.10	73.24	52.13	34.59	66.42
LEPD	76.44	79.20	33.71	34.02	67.13
SEPC	76.76	77.00	32.84	44.53	60.08
SEPD	78.14	79.44	31.33	34.99	73.88

TABLE 5.8: Performance (in terms of accuracy) of generated stories to capture persona

Model	ROUGE_L
Glocal	0.1805
MPP	0.1713
LEPC	0.1814
LEPD	0.1731
SEPC	0.1665
SEPD	0.1689

TABLE 5.9: ROUGE_L scores for the generated stories by each of our models

5.3.3.1 Quantitative Results

We perform two sets of experiments: (1) evaluating the performance of the models on capturing the personalities in the story and (2) performance of story generation. The former evaluation is performed using the pre-trained classifiers (5.3.2) on the personality dataset. We calculate the classification accuracy of the generated stories of the test set for the desired target personality. However, we need to note that the classification error of the trained models is also reflected in this result. This evaluation is done at a sentence level, i.e., accuracy is calculated over each sentence of the story (each sentence of the story has the same target personality as that of the entire story). The performance of the generation is evaluated using the ROUGE score ⁶. Although this captures the generic aspect of generation, the metric explicitly does not evaluate whether the story is generated on a conditioned personality. In the future, we would also like to look at automatic evaluation of the generated stories with respect to the incorporation of personalities.

Table 5.8 shows the results of classification accuracy for each of the five personalities. Table 5.9 shows the results of ROUGE_L evaluation. We acknowledge that there would be a deviation to this automatic score since optimizing the gold standard generation of a story from training data is not our end goal. Rather, our models use two distinct datasets and learn to transfer the traits annotated in the personality dialog dataset into the visual storytelling dataset.

Despite this, we notice that the LEPC model gives comparative results to that of the glocal model in terms of story generation. It is noticed that the LEPC model also gives a slight improvement on the classification accuracies for most of the clusters (each cluster representing a personality). However, this is an insufficient result to generalize that incorporating personality at the context level performs better than that at the word level since the inverted stance is observed in SEPC and SEPD models. We plan to investigate this further by performing ablations and examining which operation is causing these models to perform weakly. Note that the SEPC model performs the best in incorporating personality in three of the five personality types. But this model takes a hit in the automatic score. This is because our generative models are dealing with competing losses or reconstruction of classification.

⁶We use the implementation from <https://github.com/Maluuba/nlg-eval>

5.3.3.2 Qualitative Results

We present an example of the story generated by each of the models proposed in Figure 5.9. This example belongs to persona in cluster C3. The words corresponding to this cluster are highlighted with blue color in the persona-conditioned generation of the stories. The main observation is that all of the five sentences in the story contain a word relevant to *happiness* for each of the MPP, LEPC, and LEPC models. SEPC and SEPD models capture these happiness features in only two and one sentences, respectively. The glocal model does not cater explicitly to the personality, while our proposed models attempt to capture the persona tone in generation. This is observed in the fourth generated sentence in the sequence by each of our proposed models. While the glocal model uses the word ‘*silly*’, our models capture the tone and generate ‘*excited*’ and ‘*great*’. Similarly for the fifth sentence, MPP, LEPC and LEPC generate ‘*happy*’, ‘*surprised*’ and ‘*happy*’ respectively.

It is observed that in most generated stories, the language model has taken a rough hit in the SEPD model. This is also substantiated in Figure 5.9. This seems to be due to stripping away the essential word embedding features that contribute to linguistic priors or language model. This could be potentially corrected by retaining the word embedding feature as is and augmenting it with the stripped features. Having presented these results, we notice that there is significant scope for improving the generation of the story while capturing high-level persona traits in generation. The scalar manipulations that we have seen in this section seem to affect the language model to the extent that it affects the readability. This motivates our proposed work on disentangling the latent representation to transfer the personality from an external data source that guides the latent space to generate persona induced story.

5.4 N-Global Anchoring in Language Information

One of the major problems in this domain is the dearth of annotated data (including annotations for lexical level language id information) and substantial corpora to train large-scale neural models. This makes it a variant of low resource setting (Sitaram et al., 2019) with no directly available data for scaling it to larger modeling techniques. We present a novel vantage point of code-switching to be style variations between both the participating languages. Our approach does not need any external annotations such as lexical language id, thereby not N-locally relying on additional anchoring for each unit. Instead, it mainly relies on easily obtainable monolingual corpora without any parallel alignment and a limited set of naturally code-switched sentences. Utilizing this data, we rely on a discriminator that decides whether the given sentence is: (1) code-switched or monolingual and (2) naturally code-switched or randomly switched. The latent representation derived from the monolingual data is representative of each of these styles in which the generation of code-switched text is anchored at the sentence level. We propose a two-stage generative adversarial training approach where the first stage generates competitive negative examples for code-switched and the second stage generates more realistic code-switched sentences. We present our experiments on the following pairs of languages: Spanish-English, Mandarin-English, Hindi-English, and Arabic-French. We show that the trends in metrics for generated code-switched sentences move closer to real data in each of the above language pairs through the dual-stage training process. We believe

this viewpoint of code-switching as style variations opens new perspectives to dealing with this form of surface realization. There are plenty of monolingual corpora available for each of the participating languages. We present a novel standpoint to transfer knowledge from monolingual corpora without additional annotations such as language ids or parse trees. The recent advances in cross-lingual pre-trained language models (Artetxe and Schwenk, 2019; Conneau and Lample, 2019) call out for vast amounts of code-switched data. Hence, our work on automatic generation of CS text is relevant for several downstream tasks.

We propose a novel vantage point for code-switched text to be observed as a stylistic variation between the participating embedded and matrix languages. For the scope of this paper, we define the style variations between languages to be extrinsic properties such as surface lexical forms and intrinsic properties such as underlying grammar, word order etc.. We address this problem with adversarial training in two stages: (1) *Stage 1*: transfer the style of each of the monolingual participating languages into the content of the other language; (2) *Stage 2*: discriminating between the incorrectly switched and naturally switched sentences. The four styles in play here are the following: (1) l_m : matrix language style (2) l_e : embedded language style (3) l_a : incorrect/artificial code-switching style (4) l_n : natural code-switching style. Intuitively, each of the generated sentence has N-Global anchors in one of these styles l_x where $x \in (m, e, a, n)$. The goal is to traverse smoothly across these styles without affecting the content. The first stage generates negative examples facilitating the discriminative training for the second stage. This dual-stage training eliminates the need for additional linguistic annotations, such as language id used by several contemporary works. We present our results on four pairs of languages.

5.4.1 Datasets Description

Each of the participating monolingual utterances (belonging to \mathbb{M} (matrix language) and \mathbb{E} (embedded language)) are treated as two distinct styles. Note that the sentences are not aligned either at the phrase or sentence levels. We explored code-switching for four language pairs as presented in Table 5.10.

The reasons behind selecting these language pairs are multi-step. We selected Hinglish and Spanglish since they are widely spoken languages. The usage of Hindi in Hinglish is commonly romanized, bringing in a new variety to the platform. Spanglish and Hinglish thus have very close scripts as opposed to Mandarin-English where the scripts are different. While the word order for English, Spanish, Mandarin and French is subject-verb-object (SVO), the same for Hindi is subject-object-verb (SOV) and Arabic is verb-subject-object (VSO). These differences facilitate the stylistic attributes to the mixing of these languages.

5.4.2 Model Description

The two problems we address are repealing the need for annotations (such as language id) on CS data and maximizing the utilization of monolingual data. Both the issues are addressed using a two-stage generative adversarial training paradigm with a transformer-based autoencoder. The unavailability of parallel sentences is tackled by preserving the semantics of the

Language	Monolingual	Code-Switched
Spanish English	Graff et al. (2010) Budzianowski et al. (2018)	Deuchar et al. (2014)
Mandarin English	Tian et al. (2014) Budzianowski et al. (2018)	Lyu et al. (2010)
Hindi English	Mathur et al. (2018) Mathur et al. (2018)	Mathur et al. (2018)
Arabic French	Song et al. (2014) Koehn (2005)	Cotterell et al. (2014)

TABLE 5.10: Monolingual and Code-Switched Datasets used for training Stage 1 and Stage 2

original sentence of one language and mixing the attributes of the other language without disentangling the representation into these two properties. Following are the two stages involved:

Stage 1 : The embedded and matrix languages are mixed in arbitrary ways to generate CS text. This stage simply uses the corpora from each language as an individual style.

Stage 2 : The sentences generated after Stage 1 were not supervised via any real CS sentences. Hence, they are used as negative examples (with style l_a) against limited amount of CS text (with style l_n) to generate naturally switched sentences.

The architecture remains the same for both stages except for variation in hyperparameters. Figure 5.10 presents our GAN setup for Stage 1. The following subsections present the flow by instantiating for Stage 1 for readability. The same process is applied for Stage 2 with the difference of using positive and negative examples of CS sentences.

Generator The generator in our architecture comprises of transformer based encoder and decoder. In Stage 1, our transformer encoder takes in the matrix language sentence ($s_m \in \mathbb{M}$) along with the matrix language encoding or style (l_m) and produces a latent representation ($z_{m,m}$).

$$z_{m,m} = \text{TransEnc}(s_m, l_m) \forall s_m \in \mathbb{M} \quad (5.10)$$

We use this $z_{m,m}$ along with the original matrix language sentence s_m and the matrix language encoding l_m to reconstruct the original sentence s_m . Greedy decoding is performed that uses argmax which is non-differentiable to compute the loss for reconstructing the original sentence ($L_{G(\text{matrix})}$).

$$L_{G(\text{matrix})} = - \sum_{s_m \in \mathbb{M}} \log(\text{Pr}(s_m | s_m, l = m)) \quad (5.11)$$

Next, the same matrix language sentence s_m is considered along with the embedded language encoding l_e to produce a latent representation ($z_{m,e}$).

$$z_{m,e} = \text{TransEnc}(s_m, l_e) \forall s_m \in \mathbb{M} \quad (5.12)$$

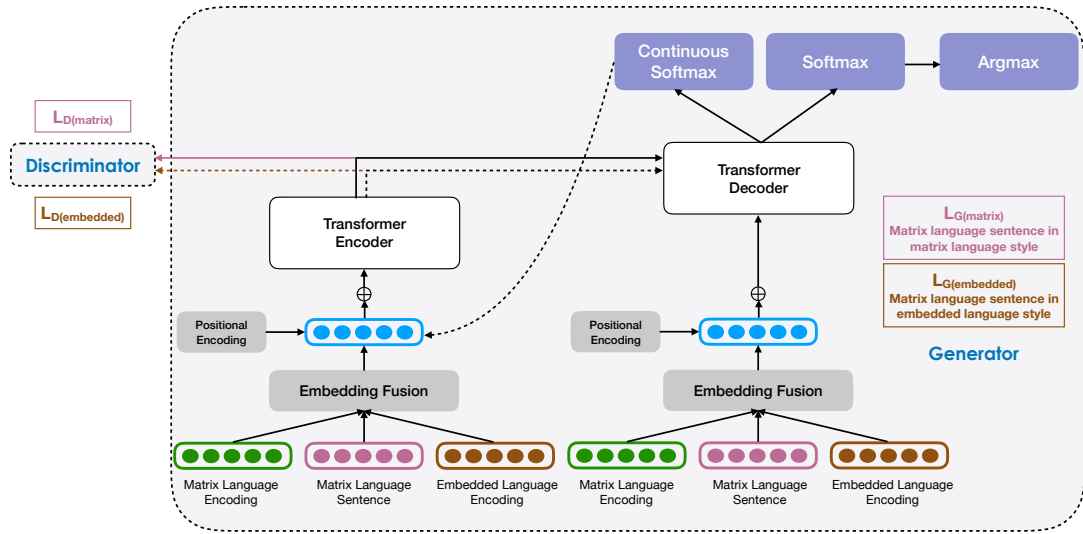


FIGURE 5.10: Transformer based GAN architecture for generating CS text. *Note: The same architecture is used for two stages. Matrix language sentence is the embedding of the text and language encoding is the embedding of the language.*

This $z_{m,e}$ is used to reconstruct the original sentence s_m . This means that the model is attempting to reconstruct the content of the original sentence while varying the style i.e., language encoding. The loss corresponding to this reconstruction is ($L_{G(embedded)}$).

$$L_{G(embedded)} = - \sum_{s_m \in \mathbb{M}} \log(\Pr(s_m | s_m, l = e)) \quad (5.13)$$

Similarly, corresponding counterparts using $z_{e,e}$ and $z_{e,m}$ are generated.

Discriminator The discriminator is a classifier that predicts whether the current distribution is closer to the original latent space or the generated latent space. The purpose of the generator reconstructing the original sentence s_m with matrix language encoding l_m (contributing to $L_{G(matrix)}$) is solely to make sure that the generator is retaining the content of the original sentence, and has no contribution towards training the discriminator. On the other hand, the generation of the sentence s_m with embedded language encoding l_e , say $s_{m,e}$ essentially establishes our end goal. In the GAN architecture, we now have two choices, i.e., sampling a sentence from: (1) original distribution $s_{m,m}$, i.e., the matrix language sentence with the matrix language encoding or (2) distribution from the generator $s_{m,e}$, i.e., the matrix language sentence with the embedded language encoding. The positive examples to train the discriminator come from real sentences, which are trained by maximizing the probability for predicting that it belongs to label m .

$$L_{D(matrix)} = - \sum \log(\Pr(m | z_{m,m}, l = m)) \quad (5.14)$$

One particular problem for training GANs in the text domain is the non-differentiable function of argmax that is performed in decoding. There are three prominent solutions to address this problem, including REINFORCE (Williams, 1992), Gumbel-Softmax (Jang et al., 2016), manipulating the latent space. We proceed with the third option by performing a continuous softmax

of the words, thus eliminating the need to perform argmax, which is described in detail here. Let the vocab size be \mathcal{V} and the embedding dimension be \mathcal{H} . Instead of discretely making a selection of the embedding over the vocabulary space to select each word, we perform continuous softmax. The final softmax layer in the decoder provides us with a vector of size $1 \times \mathcal{V}$. Multiplying this with the embedding weights ($\mathcal{V} \times \mathcal{H}$) results in $1 \times \mathcal{H}$ vectors for each word. Note that in the case of argmax, we make a discrete selection of the word, whereas, in the case of continuous softmax, we arrive at a soft representation of the weighted combination of properties of the words across different words in the vocabulary. Therefore the latter does not enforce this soft representation to be a word. This partially decoded representation now passes through the transformer encoder to arrive at a latent representation to be fed into the discriminator.

$$L_{D(\text{embedded})} = - \sum \log(\text{Pr}(e|z_{m,e}, l = e)) \quad (5.15)$$

Dual Stage Training Setup: The task of generating CS text not only entails mixing languages but also mixing them appropriately. This means that our discriminator performs two tasks of discriminating between: (1) the participating languages, owing to the asymmetry between their interactions, such as matrix and embedded languages (Stage 1) and (2) incorrectly and correctly switched languages (Stage 2). Hence we dissolve the training procedure into two stages, with each stage dedicated to one of the aforementioned tasks. In Stage 1, the sentences from $s_m \in \mathbb{M}$ are transferred to the style of l_e ($s_{m,e}$). Similarly, the converse produces the sentences $s_{e,m}$, i.e., sentences in the embedded language in the style of the matrix language. Since there is no supervision from naturally CS sentences, we delegate this responsibility to the second stage of training with the same architecture. We used $s_{m,e}$ as negative examples of CS sentences for Stage 2 of training. We have also experimented with a random subset of $s_{m,e}$ and $s_{e,m}$ as negative examples. This performed worse than the former setting since $s_{m,e}$ has the underlying grammatical structure of \mathbb{M} , thereby generating stronger negative examples for adversarial training.

Figure 5.11 presents a comparison of the condensed forms of anchoring the text with the token level language ids.

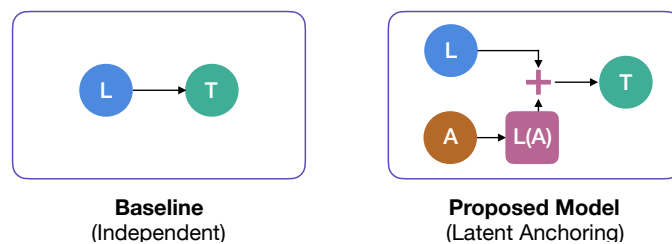


FIGURE 5.11: Comparison of the surface-form-anchoring based on discriminator for language generation

Hyperparameter setup: We used three layers of transformer encoders and decoders with a maximum sequence length of 45 words. The word embedding dimension is 256, with 300 iterations of pre-training the generator before training our GAN. In Stage 1, there is minimal overlap of vocabulary between the languages. This is contrary to data in typical style transfer

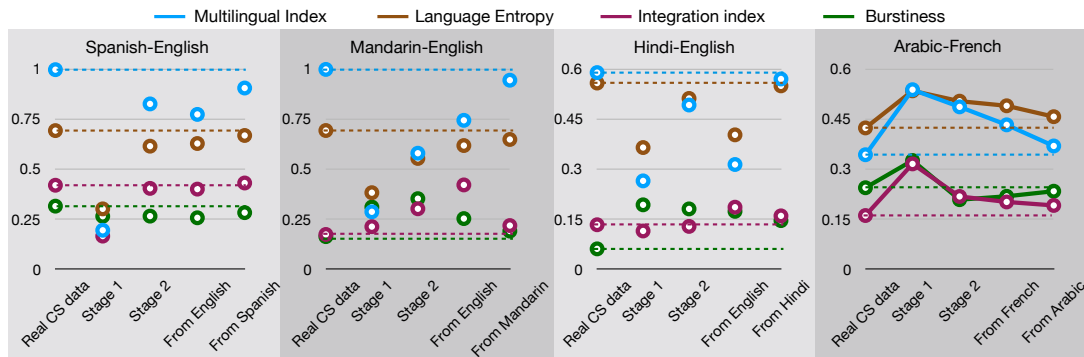


FIGURE 5.12: Trends in metrics for evaluating the generation of CS text for four language pairs in dual stage training. *The dotted line of each color benchmarks the corresponding metric for real CS data.*

datasets, which have overlapping vocabulary spaces. Hence the discriminator learns much faster than the generator in our case. To combat this, the learning rate in Stage 1 for the generator is $1e-3$, and the discriminator is $1e-4$. For Stage 2, the generator and discriminator are initialized with models learnt in Stage 1, thereby transferring the knowledge of each of the languages. However, to quickly adapt to the parameter space of Stage 2, we use slanted triangular learning rate (Howard and Ruder, 2018) with a short linear increase period followed by a longer decay period. Adam optimizers are used throughout the model. We plan to release our code, models, and generated samples upon acceptance.

Anchor Representation The N-Global anchor is incorporated in the form of style along with each sentence in the form of one of l_m , l_e , l_a or l_n . This encoding is not provided for each lexical unit but guides the entire sentence. Similarly, the penalty is observed for the entire sentence, i.e., the discriminator as described in the next subsection models the probability of the latent representation belonging to one of the above styles. Note that in each stage, the discriminator acts to classify the latent representation into two classes. In Stage 1, these classes are l_m and l_e , and in Stage 2, these classes are l_a and l_n .

5.4.3 Experiments and Results

We present our results on four pairs of languages from Table 5.10. We evaluate trends in different metrics of CS proposed by Guzmán et al. (2017) in our dual-stage training. Consolidated results are presented in Figure 5.12. Metrics that we look into are *multilingual-index*, *language entropy*, *integration-index* and *burstiness*. In Figure 5.12, ‘Stage 1’ contains generated sentences $s_{m,e}$. The model learnt in Stage 2 has options to generate from negative examples or original text of each language as source. ‘Stage 2’ contains sentences generated using negative examples from Stage 1 as source. Similarly, ‘From $\langle lang \rangle$ ’ uses the corresponding language as source to transfer style to real CS. We observe that the metrics move closer to real CS in ‘Stage 2’ as compared to ‘Stage 1’. Within ‘Stage 2’, metrics are closer to real CS data when the source text belongs to \mathbb{M} in comparison to \mathbb{E} or $s_{m,e}$ from Stage 1. We plan to explore properties of syntactic and semantic mixing in each stage in our future work.

5.4.3.1 Human Evaluation

We conduct human evaluation in the form of preference testing between groups of sentences with ten human subjects for generated Hinglish, each with ten batches of sentences. Each batch has two sets of sentences (each set having 5 sentences). The first set is from Stage 1 and the second set is from Stage 2, which are jumbled randomly. The preference testing is done at the set level. We ask the human evaluators to select a set that ‘seems more natural’. The instances are grouped lengthwise i.e, among 10 instances: *short-length* in the range of 3-5 words (2 sets), *medium-length* in the range of 6-10 words (5 sets), *long-length* in the range of 11-15 words (3 sets).

Overall, the sentences generated from Stage 2 are preferred 74% of the time. Dissecting them length-wise Stage 2 is preferred 60% in short-length, 84% in medium-length, and 67% in long-length. This shows that, though Stage 2 generation is clearly preferred for all ranges of sentence lengths, it is better in generating longer sentences in comparison to shorter sentences.

5.4.3.2 Qualitative Analysis

The following are some of the common forms of errors observed in the generated text for Hinglish when trained on the blogging data collected from [Chandu et al. \(2018\)](#) on which similar trends in results from Hinglish in Figure 5.12 were observed.

- *Gender Disagreement*: For instance, consider the sentence ‘kyunki ye scam bhi ho sakti hai’ (Meaning: because this can also be a scam). The gender of the direct object, which is ‘scam’ should agree with the inflection of the verb ‘sakti’. Hence this should have been ‘sakta’. The gender of the word ‘scam’ (which is a borrowed word from embedded language English) is unknown in the matrix language, so it would be presumed to be masculine. But this sentence used a feminine verb phrase.
- *Incorrect Case markers*: ‘agar aap bhi ye post pasand aaye toh aap puch sakte hai’ (Meaning: If this post is pleasing to you also, then you can ask). The words ‘to you’ is supposed to be in dative case which is ‘aap ko’ in the first clause of the sentence. However ‘aap’, which is in the nominative case is generated.
- *Semantically incorrect - random mixing*: Sometimes, the model also generates completely random mix of words. For instance, this is one of the sentences from the output: ‘am bahut hi achchi jankari aapko pata hi hoga’ (Loosely Translated Meaning: very good information you must be knowing). The sentence does not convey a coherent meaning. The word order of the sentence is also jumbled and does not strictly belong to either of the matrix or the embedded languages.
- *No mixing*: In some cases, the entire sentence is built from words belonging to the same language. For instance, ‘the best way to improve your application for the reasons listed below’.
- *Incorrect sub-word mixing*: Though the incorrect case markers and gender disagreement are syntactic errors, this seems to happen due to the modeling at the word level. Sub-word

level modeling of text is a promising direction to address this error category, especially for morphologically rich languages.

The category of the first two error types described above is syntactic. Our current model is purely data-driven from the surface forms. This motivates the utility of inducing syntax while generation. This also encourages the case to model sub-word level modeling, especially for morphologically rich languages. The semantically incorrect sentences seem to be generated due to the random mixing of the words from both languages. The loosely translated meaning of the sentence is not utterly senseless but, when framed in CS fashion does not make sense. In addition, there is an inherent challenge while dealing with multiple datasets that lead up to domain variation. For instance, the vocabulary or the style on social media platforms such as Twitter is very different from the domain of conversations. Although we have carefully selected the datasets to belong to similar domains, this might often not be feasible, inviting the domain invariant modeling of text.

We present a novel perspective of viewing CS as style variants between participating languages. We believe this viewpoint opens new avenues for dealing with mixed language text. The main contributions of the paper are threefold. Firstly, we eliminate the need for explicit language identification using two-stage adversarial training. Secondly, our approach transfers from bountiful monolingual resources and relies on limited CS data to generate new CS sentences. Thirdly, we present our experiments on a dual-stage transformer-based GAN model for generating four pairs of CS languages: Spanish-English, Mandarin-English, Hindi-English, and Arabic-French. In the future, we would like to compare the performance of this technique with other style transfer models and also explore the possibility of end-to-end training of both stages. We are also continuing to work on utilizing the generated data to pre-train models to perform downstream tasks.

5.5 Conclusions and Prospective Future Directions

In this chapter, we have explored the incorporation of surface realization property by anchoring at two levels: N-Local and N-Global. In §4.2, we explored two different use cases of inducing the language ids at the lexical level. The first is language modeling, and the second is speech synthesis. In the case of language modeling, we extended the capability of the state-of-the-art language model for English so as to equip it to deal with code-switched text. To enable this, we anchored each lexical unit with the corresponding language id and utilized the embedding representation of the same to be incorporated along with each word. In addition to the primary task of predicting the next word, as is typical in language modeling, we introduced a multi-task learning framework with a secondary task of predicting the language of the next word. As shown empirically, this N-Local anchoring performs better than the corresponding strong baseline.

In N-Local anchoring of language ids explicitly for speech synthesis, we presented techniques to synthesize navigation instructions in mixed language, where the instructions are rendered in one language, and the names of locations are derived from another language. Such scenarios are common in multilingual countries like India, where English is a widely-used language. We

first perform language identification and then transliterate native language words into the native script to derive appropriate pronunciation rules. We bypassed the step of mapping phones cross-lingually by using a bilingual TTS system to synthesize mixed-language navigation instructions. We performed experiments synthesizing navigation instructions with named entities derived from three Indian languages - Hindi, Telugu, and Kannada. In subjective listening tests, there was a significant preference for our proposed approach compared to a monolingual Indian English system. We also performed a listening test and open-ended interviews with drivers with low English proficiency and found a preference for our proposed approach.

In §5.4, we present a novel perspective of viewing CS as style variants between participating languages. We believe this viewpoint opens new avenues for dealing with mixed language text. The main contributions of this section are threefold. Firstly, we eliminate the need for explicit language identification using two-stage adversarial training. Secondly, our approach transfers from bountiful monolingual resources and relies on limited CS data to generate new CS sentences. Thirdly, we present our experiments on a dual-stage transformer-based GAN model for generating four pairs of CS languages: Spanish-English, Mandarin-English, Hindi-English, and Arabic-French.

Now, we will explore some of the prospective directions that are worth exploring in this direction.

1. Widening Anchor Forms: As we have seen in this chapter, all of the anchor units correspond to the language information. However, from the learning experiences in addressing the anchoring of content and structural properties, anchors can be other forms. For instance, properties of syntactic and semantic mixing such as parse tree units or part-of-speech tags can as well be explored to be anchored in. This is comparatively trivial in the case of N-Local anchoring in comparison to N-Global anchoring. Having said that, note that these are difficult to obtain in comparison to language id anchors. More often than not, the literature in determining POS tags (Vyas et al., 2014; Solorio and Liu, 2008b) and parses (Duong et al., 2017; Bhat et al., 2018; Goyal et al., 2003) rely on language identification as a first step. The other types of widening the forms may include incorporating other modalities such as corresponding speech units.

2. Two dimensional Domains: Arguably, the collection of additional code-switched data would be a significant contribution to this work. The sources for collecting code-switched data remain limited in topic and variation, and additional sources of code-switched data would be the best way to improve how well our model can generalize. This is a problem particularly for N-Local anchoring each lexical unit as the code-switched training data drives the downstream inference tasks as well. However, this problem is not limited to N-Local anchoring. In the case of dataset selection for monolingual and code-switched corpus in N-Global anchoring, this manual inspection of nearly related domains of the data for dual-stage training was crucial. While this is a pressing issue, we can also address this by combining our task at hand with domain adaptation techniques. Notice that there are two dimensions of the domains in these datasets. The first is the difference in the domains of the text itself, and the second is the

difference in languages. I believe bringing together this diverse data is important and see the potential in bringing together the techniques for code-switching and domain adaptation.

3. Spelling Variations: As we have discussed earlier, one of the major challenges in dealing with code-switched text is cross-scripting. This leads to unnormalized texts with a lot of variations. The robustness of the language model also depends on the diversity of context in which the words co-occur. Since most of the articles collected for N-Local anchoring for the task of language modeling belong to the topics of e-commerce, the latest technology, and health, this may be affected. Hence, using pre-trained word embeddings based on large monolingual corpora after aligning the embedding spaces of both the participating languages such as MUSE embeddings (Conneau et al., 2017). However, due to the non-standardized spellings in the *romanized* Hinglish text, most words that are incorrectly transliterated are not found in common multilingual embeddings, for instance, MUSE. This implies that the errors from transliteration are propagated through the subsequent parts of the model. In this perspective, a locally-spatially invariant technique might come in handy. For instance, character level CNN based model along with an LSTM at the top can be a possible direction to explore. Let $c_1, c_2 \dots c_p$ be the padded sequence of characters of a word. Each of the characters has a e dimensional embedding. Let $c_{i:i+j}$ be the concatenation of characters from i to j . Next, we take k filters of size $s \times e$, where s is the window size which in our case is 3. The feature in the CNN layer is computed and activated using a Relu, which gives a feature map of size $p - s + 1$ over which mean pooling is performed. Typically, in prior work in modeling monolingual text, character representation in addition to word-level representation seems to give good results. This latent representation is used to decode the next word. This can be explored further.

Part III

Looking Forward

6

Grounding ‘Grounding’ in NLP

The NLP community has seen substantial recent interest in grounding to facilitate interaction between language technologies and the world. However, as a community, we use the term broadly to reference *any* linking of text to data or non-textual modality. In contrast, Cognitive Science more formally defines “grounding” as **the process of establishing what mutual information is required for successful communication between two interlocutors** – a definition which might implicitly capture the NLP usage but differs in intent and scope.

We investigate the gap between these definitions and seek answers to the following questions: (1) *What aspects of grounding are missing from NLP tasks?* Here we present the dimensions of coordination, purviews and constraints. (2) *How is the term “grounding” used in the current research?* We study the trends in datasets, domains, and tasks introduced in recent NLP conferences. And finally, (3) *How to advance our current definition to align with Cognitive Science?* We present ways to both create new tasks or repurpose the existing ones to make advancements towards a more complete sense of grounding.

We as humans communicate and interact for a variety of reasons with a purpose and a goal. We use language to seek and share information, clarify misunderstandings that conflict with our prior knowledge, and contextualize based on the medium of interaction to develop and maintain social relationships. However, language has and is going to be only one of the enablers of this communication, reliant on several auxiliary signals and sources such as documents, media, physical context, etc. This linking of concepts to context is *grounding* and within the NLP context, is most often a knowledge base, images, or discourse.

This chapter is based on the following paper:

- “Grounding ‘Grounding’ in NLP” (Chandu and Black, 2020a)

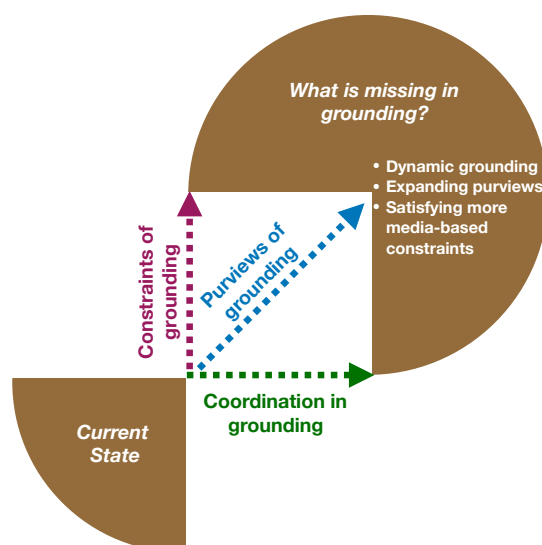


FIGURE 6.1: Dimensions of grounding – required to bridge the gap between current state of research and what is missing from grounding in real sense.

In contrast, research in cognitive science defines grounding as the process of building a common ground based on shared mutual information in order to successfully communicate (Clark and Carlson, 1982; Krauss and Fussell, 1990; Clark and Brennan, 1991; Lewis, 2008). We argue that this definition subsumes NLP’s current working definition and provides concrete guidance on which phenomena are missing from the current literature to ensure the long-term utility of our technologies.

In §6.1, we formalize three dimensions key to grounding: Coordination, Purviews, and Constraints, to systematize our analysis of limitations in current work. §6.2 presents a comprehensive review of the current progress in the field, including the interplay of different domains, modalities, and techniques. This analysis includes understanding when techniques have been specifically designed for a single modality, task, or form of grounding. Finally, §6.4 outlines strategies to repurpose existing datasets and tasks to fit the new richer definition from the cognitive science literature. The introspection, re-formulation, and concrete steps situate NLP ‘grounding’ in the larger scientific discourse to increase its relevance and promise.

6.1 Dimensions of grounding

Defining grounding loosely as *linking* or tethering concepts is insufficient for true grounding. Figure 6.1 presents the research dimensions missing from most current work.

6.1.1 Dimension 1: Coordination in grounding

The first and the most important dimension that bridges the gap between the two definitions of grounding is the aspect of coordination – alternatively viewed as the difference between *static* and *dynamic* grounding (Fig 6.2).

Static grounding is the most common type and assumes that the evidence for common ground or the gold truth for grounding is given or attained pseudo-automatically. This is demonstrated in the left box of Figure 6.2. The sequence for this form of interaction includes: (1) human querying the agent, (2) agent querying the data or knowledge it acquired, (3) agent retrieving and framing a response, and (4) agent delivering it to the human. In this setting, the common ground is the ground truth KB/data. The human and the agent have common ground by assuming its universality (i.e., no external references). Therefore, successfully grounding the query relies solely on the agent being able to link the query to the data. For instance, in a scenario where a human wants to know the weather report, the accuracy of the database itself is axiomatic, and we build a model for the agent to accurately retrieve the queried information.

Most current research assumes static grounding, so progress is measured by the ability of the agent to *link* more concepts to more data. However, the axiomatic common ground often does not exist and needs to be established in real-world scenarios.

Dynamic grounding is interactions can be achieved in two ways. The first posits that common ground is built via interactions and clarifications. The mutual information needed to communicate successfully is built via interactions including: requesting and providing clarifications, acknowledging or confirming the clarification, enacting or demonstrating to receive confirmation, and so forth. This dynamically established grounding guides the rest of the interaction by course-correcting any misunderstandings. The sequence of actions in dynamic grounding is demonstrated in the right side box of Figure 6.2. The steps for establishing grounding are a part of the interaction that include: (1) The human querying the agent, (2) The agent requesting clarification or acknowledging, (3) The human clarifying or confirming. These three steps loop until a common ground is established. The remaining steps of (4) querying the data, (5) retrieving/framing a response, and (6) delivering the response, are the same as that of static grounding. The process of successfully grounding the query not only relies on the ability of the agent to *link* the query but also to *construct the common ground from the mutually shared information* with respect to the human.

Cognitive sciences in the perspective of language acquisition (Carpenter et al., 1998) present two ways of dynamic grounding via attention: Dyadic joint attention and Triadic joint attention. In our case, dyadic attention describes the interaction between the human and the agent and any clarification or confirmation is done strictly between the both of them. Triadic attention also includes a tangible entity along with the human and the agent. The human can provide clarifications by gazing or pointing to this additional piece in the triad.

Summary: The community should prioritize dynamic grounding as it is more general and more accurately matches real experiences.

So far we discussed resolving intent ambiguities. The second kind of dynamic nature in interactions are due to the changes or evolving communicative intents and slots. These changes are often observed in scenarios where there are multiple acceptable choices presented to the speaker. Let us consider the example of a human interacting with a virtual travel assistant. The assistant presents a scenario of hotel booking and flight tickets for travel. After confirming this

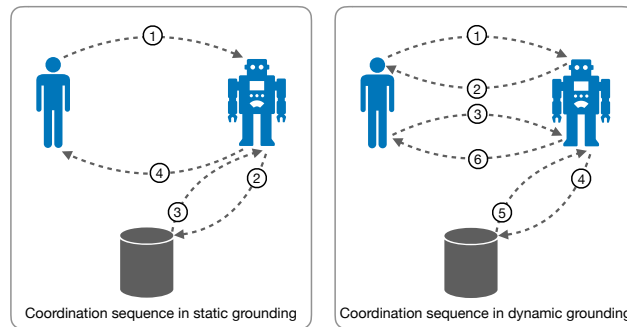


FIGURE 6.2: Coordination in grounding

and before finalizing the itinerary, if the user decides to spend an extra day at the next location, the assistant accommodating this request by updating the previous belief state is a critical step. DSTC 2 (Henderson et al., 2014) introduces a dynamic dialog state, where the users are allowed to change their goals within a domain, which in this case is the restaurant domain. In this way, there are two categories of dynamic grounding and interactive assistants need to be robust to both varying intents and confusing intents. Another layer of conversational artifacts that is missing from our current assistants is the combination of the phenomenon – misunderstanding a changing communicative goal. Dan Bohus’s thesis work (Bohus) presents one of the initial efforts in a belief updating mechanism along with error recovering strategies in a dialog between human and computer. For the remainder of the thesis, the first kind of dynamic grounding is referenced and discussed. The ability to recover from mistakes and indexing to the right turn in the conversation where the misunderstanding occurred after a corrected turn is critical to make an improved impact on real world assistants.

6.1.2 Dimension 2: Purviews of grounding

Next, we present the different stages behind reaching common ground, known as purviews. Most of the current approaches and tasks address these stages individually and independently, while they are often co-dependent in real-world scenarios.

Stage 1: Localization: The first stage is the localization of the concept either in the physical or mental contexts. This step is idiosyncratic and relates to the ability of the agent alone to localize the concept. These concepts often are also linked in a compositional form. For instance, consider a scenario in which the agent is to locate a ‘blue sweater’. The agent needs to understand each of the concepts of ‘blue’ and ‘sweater’ individually and then locate the composition of the whole unit. Clark and Krych (2004) from cognitive sciences demonstrate how incremental grounding is performed with these compositions and show how recognition and interpretation of fragments help in this by breaking down instructions into simpler ones. This localization occurs at word, phrase, and even sentence level in the language modality and pixel, object, and scene level in the visual modality.

Stage 2: External Knowledge: After localizing the concept, the next step is to ensure consistency of the current context of the concept with existing knowledge. Oftentimes, the

references of grounding either match or contradict the references from our prior knowledge and external knowledge. This might lead to misunderstandings in the consequent communication. Hence, in addition to localizing the concept, it is also essential to make the concept and its attributes consistent with the available knowledge sources. Most of the current research is focused on localizing with few efforts towards extending it to maintain a consistency of the grounded concept with other knowledge sources.

Stage 3: Common sense: After establishing consistency of the concept, a human-like interaction additionally calls for grounding the common sense associated with the concept in that scenario. In addition to the basic level of practical knowledge that concerns with day to day scenarios (Sap et al., 2020), the concept should also be reasoned based on that particular context. This contextual common sense moves the idiosyncratic sense towards a sense of collective understanding. For instance, if the human feels cold and asks the agent to get a blue coat, the agent needs to understand that the coat, in this instance, is a sweater coat and not a formal coat. This implicit common sense minimizes the effort in building a common ground reducing articulation of meticulous details. Therefore it is essential to incorporate this explicitly in our modeling as well.

Stage 4: Personalized consensus: As a part of the evolving conversations, the references in the language might evolve as well. The grounded term might have different meanings for the agent in the context with access to the history as opposed to a fresh agent without access to the history. This multi-turn process to achieve consensus makes this collective or a shared stage. In such settings, it is sufficient that the human and the agent are in consensus with the truth value of the grounded term, which need not be the same as the ground truth. This shift in the truth value of the grounded terms’ meanings often arises due to developing short-cuts for ease of communication and personalization. This shift is acceptable as long as the communication is successful.

Summary: Common ground requires jointly modeling both general and personalized contextual knowledge.

6.1.3 Dimension 3: Constraints of grounding

Communication happens via a mode or a medium in practical scenarios. The number and availability of such media have increased and facilitated ubiquitous communication around the world, presenting diversity in the mode of interaction. Motivated by this, we resurface and adapt the constraints of grounding with respect to media of interaction as defined by Clark and Brennan (1991). Here are the definitions of these constraints in the context of grounded language processing and the corresponding categorization of the majority of the representative domains in grounding satisfying different constraints.

- *Copresence:* The agent and the human share the same physical environment of the data. Most of the current research in the category of embodied agents satisfy this constraint.

- *Visibility*: The data is visible to the agent and/or the human. The domains of images, images & speech, videos, embodied agents satisfy this constraint.
- *Audibility*: The agent and human communicate by speaking about the data. The domains like speech, spoken image captions, and videos satisfy this constraint.
- *Cotemporality*: The agent/human receives at roughly the same time as the human/agent produces. The lag in the domains like conversations or interactive embodied agents is considered negligible and satisfies this constraint.
- *Simultaneity*: The agent and the human can send and receive at once and simultaneously. Most media are cotemporal but do not engage in simultaneous interaction. This often disrupts the understanding of the current utterance, and the participant may have to repeat it to avoid misunderstandings, which is commonly observed in real-world scenarios.
- *Sequentiality*: The turn order of the agent and the human cannot get out of sequence. Face-to-face conversations usually follow this constraint, but an email thread with active participants and the comments sections in online portals do not follow a sequence. In such cases, a reply to the message may be separated by an arbitrary number of irrelevant messages. These categories are usually understudied but are commonly observed online.
- *Reviewability*: The agent reviews the common ground to the human to adapt to imperfect human memories. For instance, we reiterate full references instead of adapting to short cut references when the conversation resurfaces after a while. This is to develop a personalized adaptation between the interlocutors based on the media to enable ease of communication.
- *Revisability*: The interaction between the agent and the human can index to a specific utterance in the conversation sequence and revise it, therefore changing the course of the interaction henceforth. Human errors are only natural in a conversation, and the agent needs to be ready to rectify the previously grounded understanding.

There has been a good and continual effort in formulating tasks and datasets that satisfy the constraints of visibility, audibility, and cotemporality. Recent efforts also see an increased interest in addressing copresence in grounded contexts.

Summary: Key to progress, and largely absent from the literature, is a focus on simultaneity, sequentiality and revisability.

6.2 Grounding ‘Grounding’

Having covered a more formal definition of grounding adapted to NLP, we turn our attention to cataloging the precise usage of ‘grounding’ in our research community. We present an analysis on the various domains and techniques NLP has explored.

6.2.1 Data and Annotations

To this end, we sub selected all the papers that mention terms for ‘grounding’ from the S2ORC data (Lo et al., 2020). In this way, we grounded the term ‘grounding’ in literature ¹ to collect the relevant papers. Each of the papers is annotated with answers to the following questions: (i) is it introducing a new task? (ii) is it introducing a new dataset? (iii) what is the world scope (iv) is it working on multiple languages? (v) what are the grounding domains? (vi) what is the grounding task? (vii) what is the grounding technique?

6.2.2 Domains of grounding

Real-world contexts we interact with are diverse and can be derived from different modalities such as textual or non-textual, each of which comprises domains. Our categorization of these is inspired from the constraints of grounding as described in §6.1.3. Based on this, the modality based categorization includes the following domains:

- *Textual modality comprising plain text, entities & events, knowledge bases and knowledge graphs.*
- *Non-textual modality comprising images, speech, images & speech and videos.*

Numerous other domains, including *numbers and equations, colors, programs, tables, brain activity signals* etc., are studied in the context of grounding at a relatively lower scale in comparison to the aforementioned ones. Each of these can further be interacted with along the variation in the coordination dimension of grounding from §6.1.1, that give rise to the following settings including *conversations, embodied agents and face-to-face interactions*.

6.2.3 Approaches to grounding

This section presents a list of approaches tailored to grounding. The obvious solution is to expand the datasets to promote a research platform. The second is to manipulate different representations to *link* and bring them together. Finally, the learning objective can leverage grounding. The sub-categories within each are presented in Figure 6.3.

1. Expanding datasets / annotations: The first step towards building an ecosystem for research in grounding is to curate the necessary datasets, which is accomplished with expensive human efforts, augmenting existing annotations and automatically deriving annotations with weak supervision.

New datasets: There has been an increase in efforts for curating new datasets with task-specific annotations. These are briefly overlaid in Table 6.1 along with their modalities, domains, and tasks.

¹Please note that this is not an exhaustive list of papers working on grounding as there are several others that do mention this term and still work on some form of grounding

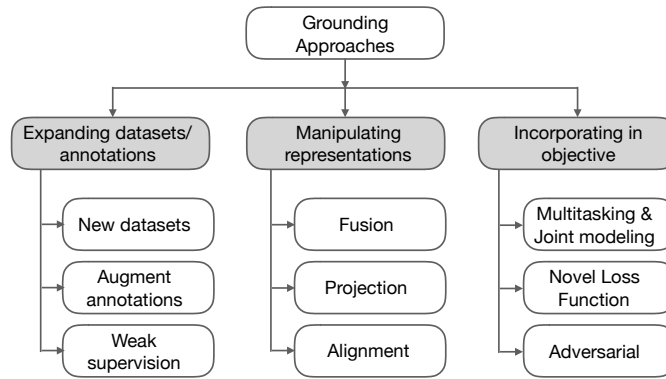


FIGURE 6.3: Approaches to grounding

Modality	Domain	Task	Work
Non-textual	Images	caption relevance	(Suhr et al., 2019)
		multimodal MT	(Zhou et al., 2018f)
		sports commentaries	(Koncel-Kedziorski et al., 2014)
		semantic role labeling	(Silberer and Pinkal, 2018)
		instruction following	(Han and Schlangen, 2017)
		navigation	(Andreas and Klein, 2014)
		causality	(Gao et al., 2016)
		spatial expressions	(Kelleher et al., 2006)
		spoken image captioning	(Alishahi et al., 2017)
		entailment	(Vu et al., 2018)
	image search	(Kiros et al., 2018)	
	scene generation	(Chang et al., 2015)	
	Videos	action segmentation	(Regneri et al., 2013)
		semantic parsing	(Ross et al., 2018)
instruction following		(Liu et al., 2016a)	
question answering		(Lei et al., 2020a)	
Textual	Text	content transfer	(Prabhumoye et al., 2019)
		commonsense inference	(Zellers et al., 2018)
		reference resolution	(Kennington and Schlangen, 2015)
		symbol grounding	(Kameko et al., 2015)
		bilingual lexicon extraction	(Laws et al., 2010)
		POS tagging	(Cardenas et al., 2019)
Interactive	Text	negotiations	(Cadilhac et al., 2013)
		documents	(Zhou et al., 2018d)
		improvisation	(Cho and May, 2020)
	Visual	referring expressions	(Haber et al., 2019)
		emotions and styles	(Takmaz et al., 2020)
		media interviews	(Shuster et al., 2020)
		spatial reasoning	(Majumder et al., 2020)
		navigation	(Jänner et al., 2018)
	Other	problem solving	(Ku et al., 2020)
			problem solving

TABLE 6.1: Pointers to curated datasets introduced to address *grounding*

Augment annotations: These curated datasets can also be used subsequently to augment with task-specific annotations instead of collecting the data from scratch, which might be more expensive.

- *Non-textual Modality:* Static grounding here includes using adversarial references to ground visual referring expressions (Akula et al., 2020), language learning (Suglia et al., 2020; Jin et al., 2020) etc.,

- *Textual Modality*: Static grounding includes entity slot filling (Bisk et al., 2016).
- *Interactive*: Though not fully dynamic grounding, some efforts here are amongst tasks like understanding spatial expressions (Udagawa et al., 2020), collaborative drawing (Kim et al., 2019a) etc.,

Weak supervision: While the above two are based on human efforts, we can also perform weak supervision to use a model trained to derive automatic soft annotations required for the task.

- *Non-Textual Modality*: In the visual modality, weak supervision is used in the contexts of automatic object proposals for different tasks like spoken image captioning (Srinivasan et al., 2020), visual semantic role labeling (Silberer and Pinkal, 2018), phrase grounding (Chen et al., 2019d), loose temporal alignments between utterances and a set of events (Koncel-Kedziorski et al., 2014) etc.,
- *Textual Modality*: In the contexts of text, Tsai and Roth (2016a) work towards disambiguating concept mentions appearing in documents and grounding them in multiple KBs, which is a step towards Stage 3 in §6.1.2. Poon (2013) perform question answering with a single database.

Summary: While augmentation and weak supervision can be leveraged for dimensions of coordination and purviews, curating new datasets is the need of the hour to explore various constraints.

2. Manipulating representations: Grounding concepts often involves multiple modalities or representations that are *linked*. Three major methods to approach this are detailed here.

Fusion and concatenation: Fusion is a very common technique in scenarios involving multiple modalities. In scenarios with a single modality, representations are often concatenated.

- *Non-textual modality*: Fusion is applied with images for tasks like referring expressions (Roy et al., 2019), SRL (Yang et al., 2016) etc., For videos, some tasks are grounding action descriptions (Regneri et al., 2013), spatio-temporal QA (Lei et al., 2020a), concept similarity (Kiela and Clark, 2015), mapping events (Fleischman and Roy, 2008) etc.,
- *Textual Modality*: With text, this is similar to concatenating context (Prabhumoye et al. (2019) perform content transfer by augmenting context).
- *Interactive*: In a conversational setting, work is explored in reference resolution (Takmaz et al., 2020; Haber et al., 2019), generating engaging response (Shuster et al., 2020), document grounded response generation (Zhou et al., 2018d), etc.,
- *Others*: Nakano et al. (2003) study face-to-face grounding in instruction giving for agents.

Alignment: An alternative to combining representations is to align them in relation to one another.

- *Non-textual modality*: Wang et al. (2020c) perform phrase localization in images and Hessel et al. (2020) study temporal alignment in videos.

- *Interactive:* Han and Schlangen (2017) align GUI actions to sub-utterances in conversations and Jänner et al. (2018) align local neighborhoods to the corresponding verbalizations.

Projecting into a common space: A widely used approach is to also bring the different representations onto a joint common space.

- *Non-textual modality:* Projection to a joint semantic space is used in spoken image captioning (Chrupala et al., 2017; Alishahi et al., 2017; Havard et al., 2019), bicoding for learning image attributes (Silberer and Lapata, 2014), representation learning of images (Zarrieß and Schlangen, 2017) and speech (Vijayakumar et al., 2017).

- *Textual modality:* Tsai and Roth (2016b) demonstrate cross-lingual NER and mention grounding model by activating corresponding language features. Yang et al. (2019e) perform imputation of embeddings for rare and unseen words by projecting a graph to the pre-trained embeddings space.

Summary: Handling different representations effectively aid improving consistency across purviews and improve modeling media based constraints dimensions.

3. Learning Objective: Grounding is often performed to support a more defined end-purpose task. This can be incorporated in the objective as follows.

Multitasking and Joint Modeling: The *linking formulation* of grounding is often used as an auxiliary or dependent to model another task.

- *Non-textual Modality:* Multitasking with images is used to perform spoken image captioning (Chrupala, 2019) and grammar induction (Zhao and Titov, 2020). Joint modeling was used in multi-resolution language grounding (Koncel-Kedziorski et al., 2014), identifying referring expressions Roy et al. (2019), multimodal MT (Zhou et al., 2018f), video parsing (Ross et al., 2018), learning latent semantic annotations (Qin et al., 2018) etc.,

- *Interactive:* In a conversational setting, multitasking is used to compute concept similarity judgements (Silberer and Lapata, 2014), knowledge grounded response generation (Majumder et al., 2020), grounding language instructions (Hu et al., 2019). Joint modeling is used by Li and Boyer (2015) to address dialog for complex problem solving in computer programs.

Loss Function: It is essential to utilize appropriate loss designed for the specific grounding task.

- *Non-textual Modality:* Grujicic et al. (2020) design soft organ distance loss to model inter and intra organ interactions. Ilharco et al. (2019) improve diversity in spoken captions with a masked margin softmax loss.

Adversarial: Leveraging deceptive grounded inputs in an attempt to fool the model is capable of making it robust to certain errors.

- *Non-textual Modality:* Chen et al. (2018); Akula et al. (2020) present an algorithm to craft visually-similar adversarial examples.

- *Textual Modality*: Zellers et al. (2018) perform adversarial filtering and construct a de-biased dataset by iteratively training stylistic classifiers.

Additional information on the categories of the models Here is a brief elaboration of the datasets presented in Table 6.1.

New datasets: The first solution is to curate the entire dataset with annotations designed for the task.

- *Non-textual Modality*: For images, new datasets are curated for a variety of tasks including caption relevance (Suhr et al., 2019), multimodal MT (Zhou et al., 2018f), soccer commentaries (Koncel-Kedziorski et al., 2014) semantic role labeling (Silberer and Pinkal, 2018), instruction following (Han and Schlangen, 2017), navigation (Andreas and Klein, 2014), understanding physical causality of actions (Gao et al., 2016), understanding topological spatial expressions (Kelleher et al., 2006), spoken image captioning (Alishahi et al., 2017), entailment (Vu et al., 2018), image search (Kiros et al., 2018), scene generation (Chang et al., 2015), etc., Coming to videos, datasets have become popular for several tasks like identifying action segments (Regneri et al., 2013), semantic parsing (Ross et al., 2018), instruction following from visual demonstration (Liu et al., 2016a), spatio-temporal question answering (Lei et al., 2020a), etc.,

- *Textual Modality*: Within text, there are several datasets for tasks like content transfer (Prabhumoye et al., 2019), commonsense inference (Zellers et al., 2018), reference resolution (Kennington and Schlangen, 2015), symbol grounding (Kameko et al., 2015), studying linguistic and non-linguistic contexts in micro-blogs (Doyle and Frank, 2015), bilingual lexicon extraction (Laws et al., 2010), universal part-of-speech tagging for low resource languages (Cardenas et al., 2019), entity linking and reference (Nothman et al., 2012) etc.,

- *Other*: More static grounding datasets correspond to tasks like identifying phrases representing variables (Roy et al., 2016), conceptual similarity in olfactory data (Kiela et al., 2015), identifying colors from descriptions (Monroe et al., 2017), correcting numbers (Spithourakis et al., 2016) etc.,

- *Interactive*: Coming to an interactive setting, the datasets span tasks like conversations based on negotiations (Cadilhac et al., 2013), referring expressions from images (Haber et al., 2019; Takmaz et al., 2020), emotions and styles (Shuster et al., 2020), media interviews (Majumder et al., 2020), documents (Zhou et al., 2018d), improvisation (Cho and May, 2020), problem solving (Li and Boyer, 2015), spatial reasoning in a simulated environment (Jänner et al., 2018), navigation (Ku et al., 2020) etc.,

In addition, there are several other techniques used to ground phenomena in real-world contexts.

A common strategy when language is involved is leveraging **syntax and parsing**. In the domain of images, Udagawa et al. (2020) design an annotation protocol to capture important linguistic structures based on predicate-argument structure, modification, and ellipsis to utilize linguistic structures based on spatial expressions. Becerra-Bonache et al. (2018) study linguistic complexity from a developmental point of view by using syntactic rules to provide data to

a learner that identifies the underlying language from this data. Shi et al. (2019) use image-caption pairs to extract constituents from text, based on the assumption that similar spans should be matched to similar visual objects and these concrete spans form constituents. Kelleher et al. (2006) use combinatory categorial grammar (CCG) to build a psycholinguistic-based model to predict absolute proximity ratings to identify spatial proximity between objects in a natural scene. Ross et al. (2018) employ CCG-based parsing to a fixed set of unary and binary derivation rules to generate semantic parses for videos.

- *Textual Modality:* Johnson et al. (2012) study the modeling the task of inferring the referred objects using social cues and grammatical reduction strategies in language acquisition. Eckle-Kohler (2016) attempt to understand the meaning in syntax by a multi-perspective semantic characterization of the inferred classes in multiple lexicons. Chen (2012) develop a context-free grammar to understand formal navigation instructions that correspond better with words or phrases in natural language. Börschinger et al. (2011) study the probabilistic context-free grammar learning task using the inside-out algorithm in-game commentaries. CCG parsers are also used to perform the entity slot filling task (Bisk et al., 2016). When applied to question answering over a database, dependency rules are used to model the edge states, as well as transitions such as the work done by using a treeHMM (Poon, 2013).

- *Other:* Roy et al. (2016) perform equation parsing that identifies noun phrases in a given sentence representing variables using high precision mathematical lexicon to generate the correct relations in the equations. Parikh et al. (2015) perform prototype-driven learning to learn a semantic parser in tables of nested events and unannotated text.

- *Interactive:* Luong et al. (2013) use parsing and grammar induction to produce a parser capable of representing full discourses and dialogs. Steels (2004) study games and embodied agents by modeling a constructivist approach based on invention, abduction, and induction to language development.

Another frequently used technique when language is involved is by leveraging the principle of **compositionality**. This implies that the meanings of its constituents determine the meaning of a complex expression and how they interact with one another.

- *Non-textual Modality:* In the domain of images, Suhr et al. (2019) present a new dataset to understand challenges in language grounding, including compositionality, semantic diversity, and visual reasoning. Shi et al. (2019), discussed earlier, also use grammar rules to compose the inputs. Koncel-Kedziorski et al. (2014) leverage the compositional nature of language to understand professional soccer commentaries. In the domain of videos, Nayak and Mukerjee (2012) study language acquisition by segmenting the world to obtain a meaning space and combining them to get a linguistic pattern.

- *Textual Modality:* With ontologies, Pappas et al. (2020) perform adaptive language modeling to other domains to get a fully compositional output embedding layer which is further grounded in information from a structured lexicon.

- *Interactive:* Roy et al. (2003) work on grounding word meanings for robots by composing perceptual, procedural, and affordance representations.

Hierarchical modeling is also applied to show effect of introducing phone, syllable, or word boundaries in spoken captions (Havard et al., 2020) and with a compact bilinear pooling in visual question answering (Fukui et al., 2016).

There is some work that presents a bayesian probabilistic formulation to learn referential grounding in dialog (Liu et al., 2014), user preferences (Cadilhac et al., 2013), color descriptions (McMahan and Stone, 2015; Andreas and Klein, 2014).

A huge chunk of work also focuses on leveraging attention mechanism for grounding multi-modal phenomenon in images (Srinivasan et al., 2020; Chu et al., 2018; Huang et al., 2019b; Fan et al., 2019c; Vu et al., 2018; Kawakami et al., 2019), videos (Lei et al., 2020a; Chen et al., 2019d) and navigation of embodied agents (Yang et al., 2020), etc.,

Some approach this using data structures such as graphs in the domains of grounding images (Chang et al., 2015; Liu et al., 2014), videos (Liu et al., 2016a), text (Laws et al., 2010; Chen, 2012; Massé et al., 2008), entities (Zhou et al., 2018a), knowledge graphs and ontologies (Jauhar et al., 2015; Zhang et al., 2020a) and interactive settings (Jauhar et al., 2015; Xu et al., 2020).

This is not an exhaustive study of all the techniques that present grounding, but are some of the representative categories. Here are more studies that perform grounding with various techniques such as clustering (Shutova et al., 2015; Cardenas et al., 2019) regularization (Shrestha et al., 2020), CRFs (Gao et al., 2016), classification (Pangburn et al., 2003; Monroe et al., 2017), linguistic theories (Strube and Hahn, 1999), iterative refinement (Li et al., 2019c), language modeling (Spithourakis et al., 2016; Cho and May, 2020), nearest neighbors (Kiela et al., 2015), mutual information (Oates, 2003), cycle consistency (Zhong et al., 2020b) etc.,

Summary: Manipulating the learning objective is a modeling capability aiding as an additional component in bringing grounding adjunct to several other end tasks across all the dimensions.

6.3 Analysis of trends

Based on the different datasets and categories of approaches from the §6.2.3, we study the trends of different phenomena. Figure 6.4 presents the trends in the development of grounding over the past decade including specific approaches (a,b), world scopes (Bisk et al., 2020) (c), and inclusivity of multiple languages (d). We also present hierarchical pie charts in Figure 6.5 to analyze the compositions of modalities and domains for these approaches.

Trends in datasets expansion: The introduction of new datasets has seen a rapid increase over the years, while there is also a subtle increasing trend in augmenting annotations to the existing datasets, as observed in Figure 6.4 (a). As we can see from Figure 6.5 (a), across all the domains, gathering new datasets seems to be prominent than augmenting them with additional annotations to repurpose the data for a new task. There seems to be a higher emphasis on the expansion of datasets in the non-textual modalities, particularly in the domain of images. A similar rise is not observed in interactive settings, including conversational data and interaction with embodied agents; which is the propitious way to bridge the gap towards a real



FIGURE 6.4: Analysis on the trends in grounding

sense of grounding. It is indeed encouraging to see an increasing trend in the efforts for expanding datasets, but *the need of the hour is to redirect some of these resources to address dynamic grounding in the coordination dimension, which is scarcely studied in existing datasets.*

Trends in manipulating representations: From Figure 6.4 (b), we note that the fusion technique has and is becoming increasingly popular in grounding through manipulating representations in comparison to alignment and projection. This is also observed in Figure 6.5 (b) with the dominance of non-textual modality. In the context of textual modality, this technique is equivalent to concatenation of the context or history in a conversation. Projecting onto a common space is the next popular technique in comparison to alignment. Similarly, we observe that the non-textual modality overwhelmingly occupies the space of manipulating representations with exceeding prominence of fusion. *Fusion and projecting onto common space currently are exceedingly used methodologies to ground within a single purview. They demonstrate a promising direction to manipulate representations across different stages to maintain consistency along the purviews.*

Trends in World Scopes: We also study the development of the field based on the definitions of the world scopes presented by Bisk et al. (2020). Based on this, the last decade has seen an increasing dominance in research on world scope 3 (world of sights and sounds). However, this is limited to this scope, and the same trend is not clear in world scope 4 (world of embodiment and action). An encouraging observation is the focus of the field in world scope 5 (social world) which is closer to real interactions in the last year. *We need to accelerate the development of datasets and tasks in world scopes 4 and 5. It is highly recommended to take the dynamic grounding scenario into account in the efforts for curating datasets in these scopes.*

Inclusivity of multiple languages: As observed in Figure 6.4 (c), research in the tasks of

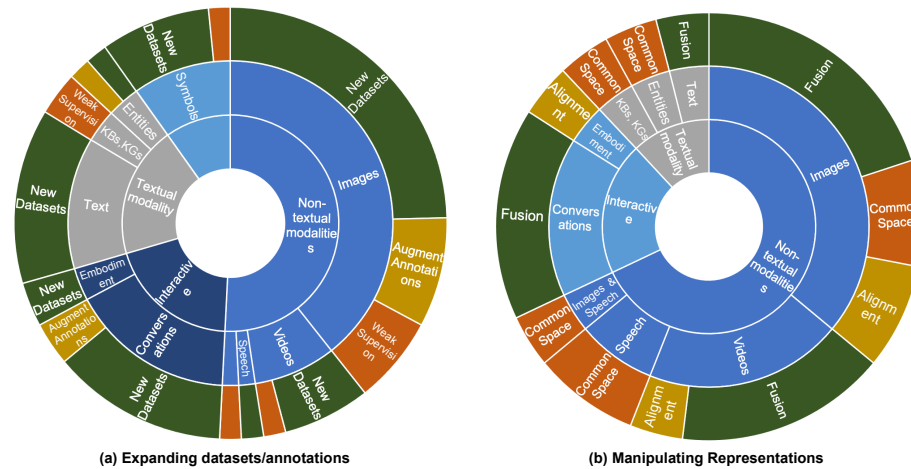


FIGURE 6.5: Analysis of Domains and Techniques

grounding in multiple languages is still catastrophically shorthanded. *The norm for benchmarking large-scale tasks still remains anglo-centric, and we need serious efforts to drive this trend to identify challenges in grounding across languages. As a first step, a relatively less expensive way to navigate this dearth is to augment the annotations of existing datasets with other languages.*

6.4 Path Ahead: Towards New Tasks and Repurposing Existing Datasets

We presented the dimensions of grounding that require serious attention to bridge the gap between the definitions in cognitive sciences and language processing communities in §6.1. Based on this, we analyzed the language processing research to understand where we stand and where we fall short with the ongoing efforts in trends in grounding in §6.2. While we strongly advocate for efforts in building new datasets and tasks considering progress along these dimensions, we believe in a smoother transition towards this goal. Hence we present strategies to repurpose existing resources to maximum utility as we stride towards achieving grounding in the real sense. In this section, we focus on concrete suggestions to improve along each of the dimensions.

Coordination: This is based on simulating interaction towards dynamic grounding using an iterative paradigm. Since establishing common ground is not integrated within the datasets, we propose an iterative paradigm that explicitly performs grounding on a common ground based on our priors. This iterative paradigm can be related to work by [Shwartz et al. \(2020b\)](#) that generates clarification questions and answers to incorporate in the task of question answering. This loop of semi-automatic generation of clarifications establishes common ground. [de Vries et al. \(2017\)](#); [Suglia et al. \(2020\)](#) also disambiguate or clarify the referenced object through a series of questions in a guessing game. This is also in spirit similar to generating an explanation or a hypothesis for question answering ([Lacinnik and Berant, 2020](#)). The process of generating an acceptable explanation to human before acts as establishing a common ground. The need

of the hour that can revolutionize this paradigm is the *development of evaluation strategies to monitor the evolution of the common ground*.

Purviews: This is based on establishing consistency across stages of grounding with an incremental paradigm. A simple solution is a modular approach where the purviews flow into the next stage after reasonably satisfying the previous stage. The popular benchmarking approaches today categorize the datasets and tasks that are similar with respect to their inputs or the outputs. This inhibits the stage-wise building to extend into different purviews. *We advocate for an orthogonal approach to this benchmarking with a pipeline model where the output distributions of different tasks are not the same but plug into one another linearly, motivated by the purviews*. The limitation of this is that the applicability of this approach is limited by the availability of the datasets in all the purviews of that domain.

Constraints: With media-imposed constraints, there is a need for a paradigm shift in the way these datasets are curated. The optimal way to navigate this problem is curating new datasets to *specifically focusing on the less studied constraints of simultaneity, sequentiality, and revisability*.

Augment with multilingual annotations: Different languages also bring novel challenges to each of these issues (e.g. pronoun drop dialogue in Japanese, morphological alignments, etc.). However, as observed in §6.3, the increase in expanding datasets is not proportionally reflected to include multiple languages. We recommend a relatively less expensive process of translating the datasets for grounding into other languages to kick start this inclusion. The research community has already seen such efforts in image captioning with human-annotated German captions in Multi30k (Elliott et al., 2016) extended from Flick30k (Plummer et al., 2015b) and Japanese captions in STAIR Captions (Yoshikawa et al., 2017) based on MS-COCO images (Lin et al., 2014). Instead of using human annotations, some efforts have also been made to use automatic translations such as the work by Thapliyal and Soricut (2020) extending from Sharma et al. (2018). Not just augmentation, but there are also ongoing efforts in gathering datasets in multiple languages (Ku et al., 2020) extending the work from Anderson et al. (2018b).

As we are ripening research with a cultivated definition of grounding, we implore the community to invest resources wisely towards achieving a practically applicable sense of ‘grounding’. More specifically, we discuss the missing pieces and dimensions that bridge the gap between the definitions of grounding in Cognitive Sciences and NLP communities. Thereby, we chart out executable actions in steering existing resources to bridge this gap along these dimensions. We also recommend systematic evaluation of grounding along these dimensions in addition to the existing linking capabilities.

7

Known Unknowns

There are known knowns. These are things we know that we know. There are known unknowns. That is to say, there are things that we know we don't know. But there are also unknown unknowns. There are things we don't know we don't know.

Donald Rumsfeld

I begin this chapter with a summary of the main contributions of this thesis. Assimilating the learnings from this work, I will venture into prospective future directions with the known context (based on findings from my thesis work), known background (related work for this context), the main research question, and potentially (un)known steps to answer this question. Finally, I discuss the broader impact on other directions or fields of study along with ethical considerations and cautionary measures to use this work.

7.1 Summary of Contributions

Anthropomorphic narrative generation in natural language in the form of stories, procedures, etc., has been a long-standing dream of artificial intelligence. Working towards this goal brings forth the need to adhere to the innate human characteristics of narratives. This includes content (relevance), structure (coherence) and surface form realization (expression). In this thesis, I introduce anchoring to these properties with local and global objectives. Anchoring these narrative properties is maneuvered not only by task-specific requirements but also by the

availability of annotated data. Moreover, steep acceleration in brewing new content every day both surmounts and impedes the need for extensive annotations. This calls for a spectrum of anchoring between supervised and unsupervised, ranging from token-level to sparse narrative-level annotations. The main contribution of this thesis is a novel two-dimensional taxonomy of anchoring these three properties by locally and globally conditioned training objectives. This framework taps into techniques for anchoring to improve narrative generation, which is detailed in Chapter 2 of Part 1. Following this, I discussed task agnostic techniques for text generation, delving into the learning algorithms, decoding strategies and evaluation methods. Then, bringing text generation and anchoring together, I detailed various methodologies to anchor different desiderata of task-specific requirements in the aforementioned text generation techniques followed by presenting categories of these methods to anchor multimodal narratives.

Consequently, I discussed anchoring each narrative property in the three chapters in Part 2. To begin with, the first property of effective narratives is the quality of being closely connected, appropriate and informative, thereby contributing to the *content* or relevance. I investigated methods to improve content in Chapter 3. Here, I first presented a dual staged method with constrained cross attention on weakly supervised anchors (derived as content units) to denoise web-scale vision and language data. These anchors also demonstrated cross-lingual transferability across five languages. In addition to empirical improvements in human evaluations, I also evaluated the quantification of these improvements from visually informative captions via cross-modal discourse coherence. I also qualitatively demonstrated their ability to generate controllable captions along with their ability to provide for human-interpretable intermediate representations to perform human-in-the-loop corrections. Introducing these content words is not a sufficient parameter to improve coherence; referring to them henceforth is also crucial. Following this, I presented a hierarchical glocal attention model to learn to introduce entities and refer to them in visual stories. In addition to evaluating automatic generation-based evaluation metrics, I also studied them with the percentage of nouns and pronouns generated in the narratives. In both the above cases, we used weakly supervised labels to inform anchors for the generation model. However, in the absence of explicit anchors, deriving missing contexts implicitly via infilling is a simple yet effective technique, especially with high overlapping contexts. I demonstrated the utility of this in training and inference to generate a context for missing images.

The second property of effective narratives is the quality of being logical and consistent following a *structure*. I discussed this in detail in Chapter 4 where I demonstrate gains from anchoring to a structural layout by scaffolding structure representation from unannotated textual recipes. These structural representations are learnt in a hard form in an unsupervised fashion from a sequence of clusters. I also demonstrated using a finite state machine to represent the state transitions in hard and soft forms. Finally, I presented models to use these derived representations in the decoder along with a hierarchical multitasking objective to visual recipes. In addition to anchoring structure locally, this can also be done at the narrative level by reordering the sentences presented for summarization. Unlike content, structure is a more abstract property, and competently representing structure across multiple domains like procedural texts, persuasive texts, etc., remains an open challenge.

The third property of effective narratives comes from the choice of surface tokens derived from underlying representation contributing to the *expression* of the text. I presented ways to do this in terms of persona and multilinguality in Chapter 5. First, when anchoring to the persona, I derived personality characterization clusters from weak supervision. I utilized them to update the generation model for visual stories with early and late fusion techniques. Second, for anchoring the language choice of tokens in code-switching, I used lexical level language ids to improve generation in multiple tasks. Starting with language modeling, I show the efficacy of using language id anchor units in encoding, decoding, and autoencoding. I also extend this idea to improve the speech synthesis of navigational instructions. Instead of using local, i.e., lexical level anchors, I worked on improving unsupervised text generation with a dual-stage adversarial method. The continuous softmax representation of the individual sentences is used as global sentence level anchors for the discriminator. For automatic evaluation, I compared the trends for multilingual index, language entropy, integration index, and burstiness. Finally, I showed results on a grouped evaluation form for human evaluations, where preference testing is carried out over groups of sentences to arrive at an average preference.

Finally, in Part 3, I presented the missing dimensions of grounding in NLP as contrasted with cognitive sciences paving the way to the future directions. Specifically, this chapter challenges the current operation of ‘grounding’ in NLP under the assumption that it is any kind of linking of text to data or non-textual modality. First, to bridge the gap with cognitive sciences, I discussed where we fall short in coordination by contrasting static and dynamic grounding with prospective methods to address them using human-in-the-loop and iterative refinement. Second, I revisited the current paradigm of lateral benchmarking to expand purviews using longitudinal benchmarking. Finally, I discussed media-imposed constraints that manifest grounding in different modalities and cues.

Learning from the findings of this work, I present some potential future directions in the next section that can be worked on as a consequence of some of the informed *knowns*.

7.2 Future Directions

Short-term:

Unpaired Visual Cross-lingual Generation:

Known Context: An often demanding expectation that promises improved performances is the availability of vast amounts of paired data. However, this is a vexatious assumption, particularly in scenarios where we scale across languages and domains and extend this to narratives.

Known Background: Prior work explored unsupervised image captioning (Feng et al., 2019), where the images and captions are projected onto a common space and are used to reconstruct each other to ensure semantically consistent texts with the image using bi-modal reconstruction. Similarly, Gu et al. (2018c) address this by learning a model in a pivot language and then translating it to the target language.

Research Question: Can anchors be leveraged as condensed information units for pivoting into different unpaired languages or domains?

Unknown: Chapter 3 discusses selective cross-attention on the anchors for improving image captions. We conducted initial experiments for unpaired captions with anchors as intermediate representations. The anchors derived from the noisy captions drop the CIDEr scores by only 8 points when image information is completely absent. Unpaired captioning (where paired data often is present in one or none of the domains or languages) is an important yet relatively under-explored problem with valuable applications. Obtaining visually informative anchors or information units, as studied in this chapter, has the potential to scale this to multiple *languages* (Thapliyal and Soricut, 2020), *domains* (Yang et al., 2019a) and *styles* (Gan et al., 2017). First, for the case of various languages, we have examined the convenience, practicality and efficacy of anchors for cross-lingual transferability to generate image descriptions. Second, for domains, biomedical image captioning caters specifically to assist physicians in diagnosing the condition of a patient. Shin et al. (2016) have used predicting MESH terms as intermediate anchors to generate captions for chest X-Rays. Here, predicting one of the granular 57 classes additionally as anchor units helped improve the captions. This tag and generate approach is the key feature to maintain factuality. However, paired data with various medical ontologies are not only unavailable but also very expensive to create both with respect to money and the availability of domain experts. Unpaired captioning techniques have immense utility here. Additionally, utilizing anchors from typologically similar languages for cross-lingual generation and a notion of close domains for cross-domain generation is worthwhile to explore.

Identifying these anchors from a large yet noisy dataset can also assist in novel object captioning (Agrawal et al., 2019) and thereby novel narratives. I conducted initial experiments on end-to-end modeling of the dual staged generation combining the anchor prediction and text generation. In principle, the motivation is similar to SkeDecoding. The main advantages that this has to offer are two-fold. First, it nearly halves the inference time of the generation model with no additional separate step of anchor prediction. Second, the anchor prediction receives more involved feedback from the loss of the decoder for generation.

Dynamic Grounding:

Known Context: In Chapter 6, I discuss the relatively less studied dimensions of grounding that sees beyond the simple linking of the non-textual modality or data to the tokens. I made initial efforts to improve a couple of these dimensions.

Known Background: Recent advancements in embodied AI also focus on clarifications and disambiguations in conversations. Multiple rounds in conversations enable both incremental building of grounding contexts and appropriate course corrections when necessary. Most earlier work primarily focused on static instructions (Shridhar et al., 2020), memory augmented neural model to enable hierarchical decision making (Nguyen and Daumé III, 2019), providing clarifications (Chi et al., 2020) and ambiguity resolution (Thomason et al., 2019; Kottur et al., 2021a). Recently, Ido Dagan’s keynote talk at EMNLP 2021 also emphasizes on the practical utility of transforming NLP tasks such as summarization, sentiment analysis etc., to interactive or dynamic settings.

Unknown - Dynamic Grounding: The premise in this thesis is the assumed truth value of the anchors and conditioning the generation on these anchors. However, a key component of real interactions is the collaborative and incremental building of the common ground for the anchors. Asking clarification questions and providing additional clarifying responses is a central tenet to achieve this. In our ongoing efforts, we are building an interactive multimodal dataset with AI2Thor (Kolve et al., 2017) for task-oriented dialogs by explicitly modeling scenes that have confusable objects attributed by various properties. Our work here aims at this posit: *common ground is achieved via interactive clarifications*. We focus on the task of grounded and situated skill learning.

There are existing works in conversational AI on static images or room-to-room navigation tasks. While assisting someone in getting from point A to B is an essential building block of our work as well, we also focus on the missing gaps that include realism and linguistic complexity with object interactions, long-horizon plans, or compositional instructions. Some contrast with the existing work is detailed here:

- CHAI (Misra et al., 2018): ‘interact’ is the main action available here. In contrast, we will have multiple and more complex set of actions and action sequences.
- VirtualHome (Puig et al., 2018): It has a third-person view. There is no clarification-based interaction. In contrast, our goal is to interact with the simulation with the intent of deriving clarifications.
- ALFRED (Shridhar et al., 2020): It has static videos. We have interactions with corrections to the current path of establishing a goal in a common ground. The similarities between them are that they are goal conditioned, i.e., task-oriented.
- R2R (Anderson et al., 2018b): This mainly focuses on navigation only. In contrast, we will have high-level actions sometimes without step-by-step instructions.
- Embodied QA (Das et al., 2018): This has template-based language for questions. In contrast, we have more natural questions (accompanied by some natural noise that human articulations include).
- SIMMC 2.0 (Kottur et al., 2021b): It has one of the tasks focusing on ambiguities in shopping scenarios in static environments. In contrast, we have diverse day to day actions in interactive virtual environments.

In this work, our main focus is on generating clarifying questions and resolving ambiguities. In the language end of learning, we aim to achieve realizations of a diverse set of instructions to achieve a common ground through coordination. In the visual end of learning, we aim to establish this common ground through correlations between visual and language representations.

There is also a growing interest in this direction owing to the practical utility and the next-generation capabilities of embodied virtual agents. Very recently, Padmakumar et al. (2021) not only introduced a dataset studying ambiguity resolution and recovery of mistakes that includes embodied conversations interleaved with actions in the environment but also provide a platform for the community to contribute towards this open goal.

The factors contributing to these clarifications arise from contextual awareness, including occupancy, relative and absolute proximity, attributes like color and geometry described in the forms of adjectival references, spatial references (absolute, relative, landmark-based), and combinations of the above phenomena. The language phenomena towards clarifications include explicit questions, implicit questions (repeating the utterance), choice from multiple options, etc., Extending the anchors from static environments to these dynamic environments requires compositional and iteratively building these anchors.

Unknown - Longitudinal Benchmarking (discussed in Chapter 6): The lateral paradigm of benchmarking made remarkable strides in encouraging model generalization. However, translating to practical use cases is often challenging due to the limitations in purviews or scope for each task. As part of the GEM effort (Gehrmann et al., 2021) for 2022, we are planning to gather a new multi-purpose dataset that caters to multiple problems like document grounded generation, multi-sentence generation, etc., in addition to being multilingual.

Medium-term:

Interpretable Evaluation:

Known Context: Automatic evaluation metrics for generation are often unreliable (Celikyilmaz et al., 2020) and are still being heavily studied. On the other hand, human evaluations are non-standardized hindering their reproducibility due to the subjective nature in human preferences (Howcroft et al., 2020) and are very expensive and time-consuming .

Known Background: The similarity between generated and gold-standard texts are often computed with n-gram overlap metrics (Papineni et al., 2002b; Banerjee and Lavie, 2005b), content-based overlap metrics (Nenkova et al., 2007) and diversity-based metrics (Zhu et al., 2018). These methods are further crippled by artifacts from *translationese* (Freitag et al., 2020) that are alleviated using diverse paraphrases.

Research Question: Can anchors be used to potentially improve the evaluation paradigm of generation?

Unknown: Evaluating discrete labels is a relatively reliable method as compared to evaluating sequences of texts using the aforementioned methods. Based on this premise, there are two ways that anchors can be used for evaluation. First, these anchor units can be reliably extracted from inputs and evaluated similar to multi-label classification like precision, recall, f-score. Please note that the limitation to this method is that we need to have prior knowledge of the what makes up for suitable anchor tokens that forms the label space. This helps in evaluating the quality of the predicted anchors. The second is evaluating the anchors from the generated output texts. This is motivated by evaluating the consistency and factuality of the generated texts. Factual consistency in evaluating summarization has been studied by answering the questions from generated summaries based on precision (Scialom et al., 2019), extending it to precision and recall (Scialom et al., 2021). Wang et al. (2020b) evaluate the consistency by building a QA model to answer questions on the source document and the summary. While these are very sophisticated methods, they heavily rely on the generated questions and the backbone QA model. In contrast, the anchor units present a potential for oversimplified representation of the content in the source documents and summary that can be evaluated without

the dependency of an external NLU model to answer questions. Evaluating the anchors also presents adaptable ways to measure the task-specific desiderata. For example, in Chapter 3, I measure the percentage of nouns and pronouns in the generated visual stories. It is in similar lines of motivation with *Lite*^{2.0} Pyramid (Zhang and Bansal, 2021) which attempts to predict the semantic content units from semantic triplet units. This is done by predicting simulation easiness using XGBoost. It maintains the advantages of retaining expensive semantic content units while balancing with the automatically extracted semantic triplet units for the low-scored half of the sentences. Both the features used by the regressor and the semantic triplet units based on SRL are interpretable. This work is very close to the motivation of our proposal to evaluate generation by the anchor units' ability to ground text.

Beyond Factual Prompting - Structure and Multimodality:

Known context: Understanding and representing the narrative property of structure i.e., coherence is very challenging. In Chapter 4, I used a hard and soft representation of transitions between clusters on unsupervised data as a form of structural representation, where the soft representation demonstrated better performance.

Known Background: Recently, language models are being explored to learn *content* or factual knowledge in the form of answering factoid questions (Radford et al., 2019), common sense questions (Sap et al., 2019), predict relations between entities (Petroni et al., 2019; Wang et al., 2021a). The discrete language prompts are generated based on pattern matching, dependency or based on paraphrases, ensembling prompts (Liu et al., 2021; Jiang et al., 2020b).

Research Question: Can LMs be used as a tool to prompt for structure and multimodal information?

Unknown: Prompting transforms NLP tasks and poses them as fill-in-the-blanks or complete the sentence kinds of problems. It is a very handy and effective way to understand what the pre-trained models have learnt. Using appropriate prompts like *TLDR*; enables summarization of long texts (Raffel et al., 2020) and language ids enables machine translation (Liu et al., 2020). It offers several advantages without loading the model with any additional parameters. Identifying the right prompts can solve more than one task, and it directly uses the outputs of the model devoid of any internal correlated representations. Most of the very recent prior work explored what and when the language models know this information (Jiang et al., 2020b,a). In addition to these facts, language models are also very coherent and logical. As the next step, I believe that exploring prompting to improve structural coherence is worthwhile. Shin et al. (2020) have created prompts automatically via gradient-based search techniques proposing Autoprompt. The gradient-based methods iteratively replace the highest-ranked token after the dot product with the gradient for the initial trigger. However, as we have seen in Chapter 4, a soft representation of structure performed better over a discrete hard representation. For this reason, as future work, I plan to explore continuous prompts to maintain logical consistency in long-form texts. Continuous prompts were used earlier in the contexts of prefix tuning (Li and Liang, 2021) in the form continuous embeddings. However, the downside of these prompts is that they are not interpretable. Besides, venturing into this recent field of study for prompting remains nascent in multimodal domains. Combining generation and classification capabilities, UNIMO (Li et al., 2021), VLP (Zhou et al., 2020a), OSCAR (Li et al., 2020b), VinVL (Zhang et al., 2021) etc., apply cross-modal modeling techniques with combinations of contrastive loss and

seq2seq objectives. In most of these techniques, object tags are used as anchor points (Li et al., 2020b) to improve learning alignments. Hence future work can use discrete or continuous prompts to probe the multimodal information in these large pretrained models.

Controllable Generation:

Known context: Training on the massive amounts of free text available online does not directly give us an explicit control to adjust attributes like style, sentiment, content, etc.,

Known Background: Adjusting the log-likelihood scores of the language model to enable better control has been explored by augmenting an additional reranking score with topic embeddings and distributional constraints (Baheti et al., 2018) and likelihood-free importance weighting (Grover et al., 2019).

Research Question: How to use anchors efficiently to control generation?

Unknown - Deriving Control Anchors: The missing piece from this thesis to arrive at full-fledged controllability. There are two ways to extend the utility of anchors to achieve this. The first is the automatic derivation of multiple relevant versions of each anchor unit. In Chapter 3, I discuss the methods for controlling generation with appropriate anchors. However, these anchors are constructed manually to explore the effectiveness of the anchors to control multimodal generation and evaluate anchor conditioned generation. Due to the selective text-as-side cross-attention mechanism, there is an increased influence of the anchors in the generated text. For the very same reason, an irrelevant anchor adversely affects the generated text (Figure 3.6). Automatically deriving both relevant and diverse anchors still remains an open challenge. Combining derived anchors with controllable generation techniques contributes towards the creative generation of text from multimodal inputs similar to the experiments in §3.2.3.4.

Unknown - Decoupling content and surface forms: The content words that the aforementioned anchors resonate in guiding texts resonate with the stylistic attributes. Chapter 5 explores techniques to induce persona or style-specific surface form realizations in visual stories. However, decoupling the content from stylistic attributes of the language remains an open challenge in NLP as they both go hand in hand on some level. The surface form of the words does not simply carry the style and is not devoid of the content.

Long-term:

Low Resource Multimodal NLP:

Known Context: With the recent successes of large pre-trained models and courtesy to the available resources online, they have gradually perforated to large multilingual pre-trained models. However, the same momentum is not picked up in multimodal multilingual models, leading to gap in low resource multimodal NLP.

Known Background: The proliferation of multimodal efforts to several other languages has two paths. The first option is by gathering new relevant datasets, and the second option is by using independent non-parallel multimodal and multilingual datasets. Coming to the first, Huang et al. (2021) gathered Multi-HowTo100M, which is a multilingual instructional video dataset and present intra-modal and inter-modal contrastive objectives along with visual pivots. However, the scantiness of labeled data for both multilingual and multimodal poses a

problem to advance research leading to the second option above. Coming to the rescue, there are resources independently for both these streams. Recent efforts have also tried to combine traditional unimodal multilingual masked language modeling along with multimodal masked language modeling, and masked region modeling (Ni et al., 2021). This study reveals that training with five languages is better than 50 languages due to inaccuracies in translated data. The translation artifacts and the visual pivots mentioned in both these streams above can be modeled in the form of anchors and techniques from unpaired captioning can be used. However, on a seemingly unrelated but heavily related note, Caswell et al. (2021) audit the quality of the enormous web-mined multilingual corpora used to train these models. It is astounding to see that about 83 corpora are mislabeled or use ambiguous language codes. Hence, a careful choice of the multilingual resources, especially for low-resource languages is imperative to keep in mind for practically reliable contributions.

Research Question: How to efficiently use non-parallel independent multimodal and multilingual resources to improve multimodal NLP?

Unknown: As Caswell et al. (2021) have studied, a macro-average of language noise demonstrates the liability per language. Per sentence level, the inferior quality of sentences, particularly for low-resource languages, goes unnoticed. In principle, several research studies have demonstrated that utilizing more than one modality improves downstream tasks like machine translation (Yao and Wan, 2020). In contrast, it is also very challenging to apply this practically where including multimodal information reduced BLEU and METEOR scores for German and French captions (Barrault et al., 2018). However, improvements are shown with visual contexts for challenging language pairs like English-Czech with distinctive word order and morphology. Follow-up studies have also demonstrated the granularity of masking as a means to corrupt the input text to improve robustness and enable the model to rely heavily on the visual contexts. Utilizing bilingual dictionaries as anchors, potential future work can focus on improving the translation in low-resourced languages using hierarchical decoding techniques discussed in Chapter 2. Firstly, Identifying these keywords as anchors enables iterative refinement of the final narration better with keyword contexts. Secondly, exploring multimodal data collection in low resource languages with coverage of semantic buckets based inter-modal phenomena. For instance, Arun et al. (2020) present a tree structure meaning representation (Balakrishnan et al., 2019) to use limited data by grouping semantically similar data points into coarse, medium, and fine-grained buckets. The reason behind this is the coverage of meaning representation along with dynamic data augmentation, therefore, percolating similarly motivated ideas for multimodal NLP is constructive.

When are hallucinations good?

Known Context: Hallucinations are unreal experiences that appear real but are only created within the remnants of the mind (or the model). Oftentimes, modeling approaches curb or rectify the model from hallucinations for legitimate reasons. In Chapter 3, we discussed how anchoring encourages the model to generate visually informative image captions devoid of these hallucinations. Specifically, §3.2.3.3 presented an analysis on the ability of anchors to increase visible captions in comparison to meta and story-like captions. However, *is regulating hallucination the modus operandi?*

Known Background: The expectation of a model to abide by the enforced prescriptions within the limits of the data to ensure loyalty and fidelity in predictions is justifiable. While this is utmost useful, hallucinations are not always harmful. Several ongoing research efforts legitimately attempt to reduce hallucinations (Rohrbach et al., 2018; Zhou et al., 2020b) as this is considered an undesired model behavior. The primary motivation behind discouraging hallucinated words is to reduce the effects of co-occurring words. Sun et al. (2022) explore explanations for hallucinated words and attempt to minimize hallucinations arising from correlations.

Research Question: In what scenarios are hallucinations beneficial, and how to model these admissible hallucinations?

Unknown: *The mark of an educated person is the ability to make a reasoned guess based on insufficient information.* Let us break this down into the two comprising components. The first is *insufficient information* that motivates the necessity for hallucination. Real-world sequential data does not often co-occur with paired textual descriptions. Replacing the missing data with substituted values for an entire component is known as item imputation in statistics. Similarly, in Chapter 3, I discussed imputation during the inference stage for generating a textual description for missing visual contexts. The insufficiency of information or missing information in the input encourages the model to learn correlations among the interacting objects to generate invisible context. The ability to *hallucinate* this information is desirable in such contexts.

Most of the vision and language tasks in the research world focus on aligning the information across modalities via image captioning, visual question answering, visual dialog, multimodal translation, etc. While understanding the relevant conjoint information is indispensable, we are overlooking the importance of understanding the *relevant disjoint* information. In most practical scenarios, we discuss complementary information ‘*about*’ what we see rather than describing the exact information of what ‘*exactly*’ we see. In an invited talk by Dr. Jason Baldridge on ‘If Bears were Bees and Cats were Researchers’, at the ALVR workshop ¹, he asserts that images and text, when combined, deliver completely different contextualized narratives. He presents this key point on the language associated with an image in comics provides new additional information but not repetitive information. Iyyer et al. (2017) also previously collected a dataset of comics and performed analysis to support how neither text nor image can independently convey the story in a comic book but rather contain complementary information. In other words, reasonably hallucinating connective inferences across multiple plot units are required to better leverage the narrative’s context. The second component is *making an educated guess*. Utilizing common-sense to derive occluded contexts in images is, in fact desirable in some cases (for instance, in a caption ‘a woman sitting on a bench in a park’, mentioning the bench despite the bench not being visible demonstrates necessary deduction skills). Hou et al. (2020) jointly model common-sense and relation reasoning to predict visibly unaware or occluded concepts from knowledge graphs. Reasoning behind or generating rationales for the interactions between participants in a visual context is crucial step to interpret the visibly unavailable context (Zellers et al., 2019; Yin et al., 2021) This also improves describing the contextual relationships between the objects and events in the media. Hence, extrapolating the

¹<https://alvr-workshop.github.io/>

world beyond visible pixels can merge this hallucinated yet necessary context for higher-order cognition.

7.3 Broader Impact

(1) Specialized Domains:

Space Technology:

We have heard or used the phrase ‘this stuff is not rocket science’ several times. Well, then can NLP contribute towards this perceivable complex rocket science or space exploration? AI venturing into assisting space technology has several applications ranging from architecture of missions to astronaut assistants. Science fiction has always portrayed optimistic goals and utility of *Astronaut Assistants* with examples like Tars, Case, etc. These assistants can perform gargantuan calculations and sift through inconceivably vast information with a natural language interface to provide concise and useful information. First, assistive technology like Daphne (Bang et al., 2018) to design space missions can reduce the information overload of complex phenomena. It interacts with the user to provide information in the form of answers to specific questions in natural language and a visual interface. It also provides coherent feedback to a proposed design that requires the capabilities of long-form multi-sentence coherent texts to make it sufficiently detailed. Second, with hostile temperatures and living conditions, teaching robots to perform dexterous tasks outside the space craft or the space station can have tremendous implications in reducing dangers to human lives. While understanding and following natural language instructions to fix equipment is one-half of this, generating the description of the events or issues along with clarification questions is crucial in unprecedented scenarios. Most of the data also includes visuals sent to the astronauts and the ground station making it critical to understand domain-specific multimodal information. For instance, NASA is working on building a robot assistant named *Robonaut* aboard International Space Station (ISS) that can carry out risky jobs ². A similar effort known as *Crew Interactive MOBILE companion* astronaut assistant (CIMON) ³ is an emotionally intelligent robot with voice control systems along with social capabilities to decrease the stress caused due to isolation in long-term missions. It also has the capabilities of documenting experiments. Presenting a gist of meaningful results from a cogent understanding of vast logs would also require anchoring the compilation with a logical structure and selecting the appropriate content units. Third, similar to using GPS to build navigation systems on the Earth, we can extrapolate this to building a similar system to navigate extraterrestrial bodies. In collaboration with Intel, NASA developed a virtual MoonMap ⁴ from millions of images gathered by the Lunar Reconnaissance Orbiter (LRO). Anchoring the images to visually similar bodies to plan routes interpretable by the astronauts assists them in having better control over the exploratory navigation. Our quest to evolve as interplanetary species can be fulfilled by working hand in hand with robots. Trusting this technology is the key to building a cooperative relationship.

Health:

²<https://robonaut.jsc.nasa.gov/R2/>

³[https://en.wikipedia.org/wiki/Cimon_\(robot\)](https://en.wikipedia.org/wiki/Cimon_(robot))

⁴<https://www.youtube.com/watch?v=nr5Pj6GQL2o>

NLG systems can vastly contribute to the widespread utility of automatic and assistive technologies in biomedical and health domains. Proliferating this literally life-giving and life-changing information to the hands of regular people, i.e., non-domain experts can reap the benefits of medicine to the fullest. These domain-specific information resources are archived and provided as a digital repository in PubMed Central⁵. As most of these open-access full-texts and abstracts are from scholarly articles, simplifying and presenting them in regular natural language is a critical bottleneck. Coupling this with multimodal information like images from radiology scans, etc., doubles down on this issue. Based on the use case, medical imaging is performed by various methods like X-Rays (images of structures like bones), CT Scan (images of cross-sections including bones and soft-tissues), MRI (detailed images of organs and tissues from magnetic fields and radio waves), Ultrasound (images of organs and structures), PET Scan (images of functioning of tissues and organs with radioactive drugs called tracers), etc., The diverse categories of conditions they are used to detect including epilepsy (PET Scan), monitoring pregnancy (Ultrasound), tumors (MRI, CT Scan), bone fractures (X-Rays) mandate the diversity in the information provided by these various techniques. There are few and distributed efforts in understanding these across different types of images. For example, [Pelka et al. \(2018\)](#) present a dataset derived from PubMed called Radiology Objects in COntext (ROCO) and study the inter-dependency and synergy between the visual components and the textual representation of the semantic relations between these components. Anchoring the content of related conditions from one of the domain of medical imagery and generating captions in another is immensely useful to retrieve information for evidence-based medicine. Secondly, complex language makes it difficult to be understood by non-native speakers, people with intellectual disabilities, and language-impaired people such as aphasic and dyslexic people ([Glavas and Stajner, 2015](#)). Compounding on this, the requirement of domain expertise makes it even harder to understand scientific medical documents. Researchers have made strides in lexical and syntactic simplification of medical documents ([Koptient et al., 2021](#)). However, gathering paired simplified and complex text and medical images scaled across different categories of images mentioned above is not a practical solution. Applying the unpaired captioning methods discussed in this thesis shows a promising direction to approach this problem. This provides varied controllability of the simplicity or style of the text. Finally, extending this to a sequence of sub-figures, [Subramanian et al. \(2020\)](#) release a dataset with manually annotated sub-figures and sub-captions. By adopting techniques from generation from a sequence of images in this thesis, we can potentially envisage generating sequences of text for a sequence of related images and also a description of the entire portfolio of a patient as the case progresses from multiple scans. Anchoring this to a structure of the case progression helps medical practitioners draw insights from the patient's history.

Legal Sector:

[Zhong et al. \(2020a\)](#) present a comprehensive overview of the applications of NLP in legal AI, including matching similar cases, summarization, intended harm detection, etc. Similar to the implications in other sectors, AI is also remodeling the legal sector with instrumental contributions from NLP. Firstly, conducting comprehensive background research is a critical yet extremely time-consuming process due to domain-specific knowledge and legal jargon. Translating day-to-day language to *legalese* and vice-versa can prove immensely helpful in

⁵<https://www.ncbi.nlm.nih.gov/pmc/>

rummaging through volumes of documents. The two critical components to achieving this are controlling the domain-specific language complexity and anchoring the relevant documents with appropriate proportionally commensurate content. Secondly, automating the generation of routine legal documents is immensely helpful both for lawyers and their clients. However, ambiguity and obscure language usage can lead to obtuse and unintentional confusions in contracts or other legally bound documents. Enabling appropriate structure to maintain a logical coherence throughout the document is essential. The simplest approximation to this is, of course, generating a fill-in-the-blank template to maintain coherence. The relevant fields are usually filled with answers in an interactive questioning and answering. For example, *DoNotPay* is a mobile application that fights inaccurate parking tickets, files unemployment benefits while also assisting in spam filtering of emails, etc., using this technology. Similarly, *Contract Express* of Thomson Reuters partners with lawyers to help in automation. Some products like *Specifio* and *TurboPatent* have also contributed towards streamlining automated documents for filing patents. Template or slot filling is not scalable to more complex scenarios, and this provides an opportunity to power our existing NLG systems to draft these documents. Thirdly, automatic contract reviews ease the process of fact-checking in one or multiple related documents. It is a hierarchical process of assessing provisions at granular levels in comparison with other successful contracts in the past. Specific content anchors on bribery or percentage shares, etc., can be verified against existing documents to suggest deemed corrections. For example, *Kira Systems* provides for verifying the presence of pre-defined provisions across various types of contracts. While the automatic generation of documents has pragmatic use cases, orthogonally, understanding legal documents by interacting with a chatbot is also where NLP can tremendously contribute. *Norton Rose Fulbright* built by using IBM Watson as the underlying systems assists in understanding privacy documents. Understanding the documents and answering in user-understandable language is unquestionably useful.

(2) Entertainment:

Digital Entertainment:

AI is already taking preliminary steps in the realm of movies and TV shows. Recently, Netflix has introduced interactive content⁶, and several other shows⁷ that presents a variety of choices in the scenes and characters thereby fabricating a story as we watch. The reactions of various characters in branching narratives and ‘what if’ plot points⁸ bring them closer to the audiences by enhanced engagement. Most of these efforts are now made in discrete choices between pre-defined options, while there are a few efforts towards more natural preferences in the forms of natural language text or speech. Anchoring these branching narratives to the content in plot points while maintaining the consistency of the characterizations and personas is the key to building a believable plot.

Branching Storylines for Games:

Role-Playing Games (RPGs) are one genre of games that have held the gaming community’s attention through a test of time and have seen a resurgence in the last decade. They involve a

⁶<https://help.netflix.com/en/node/62526>

⁷https://en.wikipedia.org/wiki/List_of_interactive_movies#Pre-1970s, <https://www.imdb.com/title/tt8038720/>

⁸<https://www.disneyplus.com/series/what-if/7672ZVj1ZxU9>

player assuming the role of a character and living out their story. A prevalent phenomenon in most RPGs today is the possibility of different endings based on the character's conversational choices throughout the game. As a reference, in the popular game *Witcher 3*, enumerating the different conversation choices lead to corresponding consequences⁹. As seen here, in the final act, encouraging Ciri, who is a deuteragonist in the game, guides the player to a positive ending scenario, while not doing so leads to a negative ending scenario where she dies. Similarly, each conversation choice the player makes in a certain scenario can help build the backstory and the persona of the character and guide them towards an ending compatible with their persona. There are two primary problems in this design. The first is that these endings and choices are predetermined and are countably finite in their combinations. The second is that there are only a few critical conversational choices that, in effect impact the paths to an ending. Due to both of these issues, the game can be fundamentally represented as a decision graph that is static. An example can be seen here for the game, *God of War*¹⁰. In this graph, the nodes are the characters, and plot points serve as the content, and their organization in the story timeline serves as the structure. The action choices made via conversations throughout the game realize the paths connecting these nodes to offer the player creative control over the storyline, thereby making it more interesting. However, the predetermined choices in the conversations are neither creative nor accommodating of free-form language, making the game stale after just a few playthroughs. Therefore, anchoring the free-form conversational choices to the appropriate edges in the graph can dynamically construct the storyline for the game by making the experience of navigating the game more natural to the player. In addition, the multitude of paths traversing the graph quenches the creative thirst of a player, thereby making the game more enjoyable for multiple playthroughs as it is not constrained by the discrete options but is open to the language creativity of the person playing it.

(3) Education:

Teachers constantly attempt to improve the way lessons are delivered to the students with a special focus on improving the engagement of the students by developing instructional social simulations (Emonts et al., 2012; Johnson and Zaker, 2012), developing interpersonal teaching relationships (Sagae et al., 2012). The two primary motivations to improve this engagement are (1) student-centric interactive learning, (2) online and accessible education. The current efforts in the confluence of NLP and education are primarily along answering (Clark et al., 2016; Li and Clark, 2015; Clark, 2015) and generating elementary science topics or facts (Rus et al., 2007; Jia et al., 2021). But venturing into multimodality and controllable generation opens up a whole new world of interactive learning experiences. Most learning experiences for students are set with rigid practices like reading, memorizing, and regurgitating. Listening to a lecture in class is mostly auditory learning. Similarly, reading a textbook is also auditory as we listen to our internal voice as we read. However, several studies have shown the effectiveness of multimodal education (Hassett and Curwood, 2009; Unsworth, 2008) kindling the interest and retaining the knowledge or skills taught in the classroom. These studies revealed that recalling and understanding lessons is made easier when multiple senses are brought together. This also caters to teaching visual or kinesthetic learners (Begel et al., 2004). The different learning styles range across modalities, including visual, auditory, reading, writing, and kinaesthetic methods,

⁹https://witcher.fandom.com/wiki/The_Witcher_3_decision_checklist

¹⁰<https://twitter.com/corybarlog/status/1119846983252893696/photo/1>

also known as the VARK framework (Ibrahim and Hussein, 2016). Türkay (2016) also present the effectiveness of whiteboard animations like VideoScribe, GoAnimate, PowToon, etc., for teaching physics concepts. Understanding these modes better is the key to building student-centric learning platforms. This requires building teaching agents capable of constructing a coherent narrative for explaining a concept. A good benefit also comes from tailoring the topics based on the needs of the students. Secondly, accessible education has been the ambition of several non-profit volunteers, political parties, etc., MOOCs (Massive Open Online Courses) provides affordable and flexible ways to deliver quality education diversifying from learning new skills like home organization, career-advancing skills etc., However, maximizing the utility of scaling this is limited by socio-culturally diverse learners (Gillani et al., 2014). DuoLingo application (Mayhew et al., 2020) worked on developing acceptable paraphrases of language and translations . The applications of this nature can benefit from controlling the content while not losing out on the meaning of the generated paraphrases. At the same time, personalizing the style of the generated paraphrases can avail the advantages from controlling the surface forms. They also personalize the structure of the course based on the learning speed or requirement of the learners by laying a plan out from the beginner, intermediate, to advanced lessons. With the submergence of our lives in the recent pandemic, educators and technology have stepped up their roles in providing this education online effectively, thereby bringing us closer to the dream of accessible education, making it more necessary than ever.

(4) Collaborative Authoring:

Co-authoring Books:

Asynchronous collaborative writing among multiple writers only has largely perforated among us with the help of tools like EtherPad ¹¹, Google Docs ¹², BookType ¹³, Overleaf ¹⁴, etc., However, extending this collaboration to suggestions from bots has several non-trivial challenges. One such crucial aspect in collaborative writing is externalizing the process of determining the plan and organization of the document is often helpful to ensure agreement between the bot and the writer. Adopting the findings in this thesis from generating a layout or structure of the document can help in coordination. The assessment from the open-ended interviews conducted by Clark et al. (2018c) for machine-in-the-loop short story writing and slogan writing revealed that it is enjoyable and it allowed them to write in a judgment-free environment. Clark and Smith (2021) monitor the revisions made to the suggestions to study the preference between suggestions from two models. In a similar vein, leveraging the revision history is a proven way to train NLP models for providing suggestions in not only providing suggestions in collaborative writing but extrinsic NLP tasks as well (Ferschke et al., 2013). Specifically, the summaries are anchored around the highly persisted sentence across revision histories that is considered significant content in the article (Nelken and Yamangil, 2008). Similarly, Yatskar et al. (2010) utilize adjacent revisions to learn paraphrases and text simplification. Beyond simple surface-level suggestions, generating content and structure anchors for the entire narrative can provide fine-grained local suggestions on the plot points or overall narrative level global suggestions based on how the story is expected to end. These suggestions also augment the

¹¹<https://etherpad.org/>

¹²<https://www.google.com/docs/about/>

¹³<https://www.sourcefabric.org/software/booktype>

¹⁴<https://www.overleaf.com/>

creativity of the writers. The usefulness of the suggestions depends on the authors and need to be tailored to their writing style, resulting in a tradeoff between suggesting surprise elements and consistency with the story arc so far. Therefore, applying control over surface forms has a great potential to adapt to individual writers. Collaborative authoring of books and novels to generate new ideas along with textual description is being used by contemporary authors like Robin Sloan¹⁵. The existing resources, facts, and characterizations can be utilized to generate new or alternate storylines. These plot points are the anchors around which the story can be tailored. One could generate memes or stories with images with personalized information. This can alter the core message of the story while leveraging the existing characterizations. Commercially, in addition to correcting grammatical errors and spellings, Grammarly also helps in writing patterns following a required style and tone. It also mentions explanations like certain framing makes the text sound ‘more confident’, ‘affirmative’, etc. Due to these reasons, this tool is also adopted by the scientific community, where D&I initiatives have circulated free promotion codes to help student authors to improve their scientific paper writing skills.

Automated/Assisted Journalism:

In a world with constant information influx, being well-informed is one of the basic necessities. Journalism is subject to issues bred by timeline recency, public interest, demographic dominance, and personalization. AI agents assisting in automated journalism for the long tail of events can address all these issues by sharing tasks and cognitive load from humans. However, this is a very responsible role demanding trustworthiness as the effect is mass communication. Most NLP tools journalists use include topic modeling, entity recognition, text classification, and sentiment analysis, etc., Here are some of the challenges that journalists face in the process of building a story. First, the evidence or supporting documents arrive in a mixture of modes. Stray (2016) mention that on an average, journalists study about 4k documents in the form of physically printed, scanned papers that need OCR, online documents, AV tapes, etc., Distilling the relevant content from these overwhelmingly large sources of information and generating an ingestible summary in natural language is very useful. Second, journalists are often looking for evidence or patterns for a pre-cognizant subject. This implies that search capabilities are more important than exploring vast numbers of documents. Oftentimes, these concepts may be eluded from traditional search capabilities such as keyword or embedding search. A common example is identifying patterns and indications of the abstract concept of ‘corruption’. Anchoring the concept to various forms of realizations is crucial to draw pieces of evidence for such concepts. Third, assistive technology to build network sketches or maps (similar to entity skeletons discussed in Chapter 3) helps draw connections across superficially disconnected yet connected content, including organizations, events, and people. *Storytelling in entity networks to support intelligence analysts* Hossain et al. (2012) use connected cliques of entities to study the skeleton patterns to study the relationships between them. These are not for data visualization tools, but rather they are more like conceptual tools enabling the journalists to think through the story. García et al. (2007) integrates journalistic standards with ontological and meta-data standards to improve such representations. Stray (2019) surveys the AI techniques used for journalism and where the difficulty of amortizing the initial costs lie. They mention that the problems or scenarios described are often very unique to individual stories being covered. However, applying our techniques to identify standard patterns of structures across

¹⁵<https://www.robinsloan.com/notes/writing-with-the-machine/>

genres of articles can assist journalists in fitting the information in these adapted templates while providing them the freedom to adjust according to the case. Finally, being responsible and anchoring to a socially responsible style of writing is necessary to avoid the possibility of any aggression or provocation. Enculturing machines with social norms and conventions to be sensitive towards human values and sentiments is fundamental while generating mass-media texts.

(5) Augmented or Virtual or Mixed Reality:

Virtual and Embodied Beings:

Virtual beings are embodiments of AI serving as the consciousness of some of our beloved or imaginary characters. They bring together the latest advancements in graphic design, computer vision, NLP, speech technology, and many more areas, envisioning a future with characters that look and sound nearly like humans, being a part of our everyday lives. Several such virtual beings are already sharing the internet space with us, such as Lil Miquela¹⁶, who is a virtual fictional influencer that dawned as an Instagram profile and rapidly gained a huge following. She markets several fashion brands and also endorses popular brands such as Calvin Klein and Prada. Similarly, Mica¹⁷ from Magic Leap is a Mixed Reality AI assistant, and Lucy¹⁸ from Fable is a virtual being pushing the frontiers of this technology in real-world applications. Such virtual beings need the ability to tailor the content of the interaction based on the age, context, and preferences of their audience. The ability to anchor these preferences is imperative for Lucy to narrate children's stories based on their favorite characters or themes to kids and describe factual news to a more mature audience. The difference in the varieties of narratives described above leads to the need for anchoring the relevant characters, events, and the appropriate structure based on highlighting the main event and going into the details in a news article versus introducing the characters and their backstories leading to the main event. With several such advancements, interacting with virtual and embodied agents such as Amazon Astro, Jibo in a situated context, i.e., enveloped in multiple modalities that have been a fantasy, is moving slowly towards reality. For instance, personal assistants equipped with real-world skills can build rapport and assist blind people in activities such as cooking, shopping, etc., In tandem, other advancements include reconstructing the social media presence (Eter9¹⁹) or an interactive re-creation of a deceased person²⁰. Venturing into the development of such agents being an uncharted territory in real-world consumer interactions also brings with them a myriad of ethical conundrums that need to be thoroughly considered.

Tourism:

Right from the advent of augmented reality, tourism has always been an industry its applications could fit right in and enrich the human experience with an annotated perception of the surroundings. Tourism, as an industry, has always been about selling an experience, and AR has shown great promise in pushing the limits of this experience, breaking the communication barriers, and making it both fun and accessible. As an example, AR mobile apps such

¹⁶<https://www.instagram.com/lilmiquela/?hl=en>

¹⁷<https://www.magicleap.com/news/op-ed/i-am-mica>

¹⁸<https://fable-studio.com/virtual-beings>

¹⁹<https://www.eter9.com/>

²⁰<https://www.youtube.com/watch?v=uf1TK8c4w0c>

as CityGuideTour²¹, ARCity²² that use object recognition to offer information about places of interest. AR glasses have shown the promise of providing a more immersive experience by the use of AI agents interacting with users, which eliminates the need for a tour guide. Despite the advances, a gap in this scenario is that the descriptions that accompany the landmarks (content) are static and are usually just excerpts taken from a common online source opening a tremendous potential to anchor content and personalization. This brings us to an AR assistant that interacts with people, guiding them through areas whilst generating descriptions that are anchored to themes like history, recent events, fun trivia about the place based on the interests of the individual. It would provide people with a much more personalized experience of the place they are touring.

7.4 Ethical Considerations

The positive or negative reverberations of studying multimodality and controlling the generation in downstream applications are momentous. Therefore, drawing nearer to this goal mandates responsible and careful considerations on their ethical implications. In this section, I touch upon how anchoring can play a role in fairness, generating more balanced datasets, etc.,

Fairness and Biases:

Explicitly providing attributions to describe people or things or concepts exclusively for *non-prototypical* scenarios challenges modeling fairness in NLP. Leaning into prototypical central notions assigns membership that is defined operationally by the judgment of what a good member means by people (Rosch, 1975; Rosch et al., 1976) along with perceptual symbol systems for archetypes (Barsalou et al., 1999) leading to unintentionally excluding members not fitting into the prototypical framework. These archetypes proliferate from cognitive biases (Greenwald et al., 1998) to data biases rendering their implications on our modeling predictions. A specific case of such biases is the reporting bias, which is the phenomenon of the frequency with which events/tokens are mentioned in documents not mirroring the frequency or degree of their occurrence in the real world. For instance, common examples studied in NLP and vision research is biased gender representation in documents (García et al., 2014; Glott et al., 2010; Jia et al., 2016), stereotypical associations (Bolukbasi et al., 2016), criminal attributions (Manzini et al., 2019) in word embeddings. They can be systematically studied using word embedding association tests (WEAT) (Caliskan et al., 2017). However, identifying these biases is only half the solution, while mitigating them, shoulders the rest. Gonen and Goldberg (2019) show debiasing methods help mask out systematic biases to some extent but do not actually make the embeddings devoid of such biases. To this end, as discussed in §3.2 of Chapter 3, extracting entity skeletons and using semi-automatic human in the loop corrections can mitigate these biases to some extent. This technique is particularly useful for languages with grammatical gender and with rich morphological agreement (Zhou et al., 2019). In such languages, there are often non-trivial morphological interdependencies when specific aspects like gender or number changes. So, surface-level replacement techniques often fail to accommodate these

²¹<https://www.cityguidetour.com/>

²²<https://www.wearvr.com/apps/arcity-ar-navigation>

levels. Hence, anchors provide a better handle to provide interpretable control as the anchors are in natural language to perform human-in-the-loop corrections.

Caution – Evolution of biases: The biases prevalent in public discourse are not static and evolve with socio-cultural circumstances. For instance, [Garg et al. \(2018a\)](#) study the correlations between several such correlated transformations, including gender bias manifestations in articles with women rights movements and notes on Asian biases across decades. While anchoring the content to control these biases is beneficial, having static control is harmful. It is important to acknowledge that this controllability in text generation, like any other model development, adapts to these ever-evolving biases with pirouetting social climates. Additionally, the enormous potential of a pre-trained model, albeit being trained on vast amounts of data, is capped by the circumstantial data until that time it is trained. So, anchoring or controlling the generation to mitigate these biases is the key to utilizing these models to their maximum capacity instead of retraining from scratch.

Generating Balanced Datasets and/or Data Augmentation:

As discussed earlier, one of the contributing factors for biases being baked into our models is data bias. Despite making explicit efforts to mitigate such data biases, ensuring the appropriate representation of the included sectors is non-trivial. For instance, despite including specific population sectors in the data, they might not be portrayed in a positive or even neutral light. As a result, such data might have amplified damaging effects in model predictions. These kinds of unbalanced datasets have downstream effects in tasks like coreference resolution ([Webster et al., 2018](#)) machine translation ([Prates et al., 2020](#)), sentiment analysis ([Kiritchenko and Mohammad, 2018](#)). Therefore generating more balanced datasets is crucial for preventing the models from amplifying these effects. Similar to cultural aspects, such implications are also associated with named entities ([Shwartz et al., 2020a](#); [Prabhakaran et al., 2019](#)). Parenthetically, studies have shown identity terms leading to increased yet unintended toxicity scores²³ on public forums. Text generation has the potential to address this by generating more balanced datasets and utilizing this for training. Nonetheless, powerful text generation models like GPT-2 are no exceptions to falling prey to such biases from prompts ([Sheng et al., 2019](#)). In such circumstances, utilizing anchors to control the generation of such counterexamples-based datasets is very useful to move forward. Controlling specific aspects defying the balance is one way to approach this. Often, these datasets are curated from human annotators to counter the pre-cognizant notion of ([Rudinger et al., 2018](#); [Dawkins, 2021](#)). For instance, balancing training data is done by selecting or creating examples that contain identity terms corresponding to both toxic and non-toxic labels ([Dixon et al., 2018](#)). However, such data might not always be easily available to sample from, especially in the defied category. Hence anchoring the generated textual samples to multiple labels in a balanced and harmonized way can help alleviate the problem of imbalanced datasets.

Secondly, scaling for a culturally diverse annotation pipeline for all tasks and languages is not easy to accomplish and was found to be a practically overlooked aspect in datasets as well. [Costa-jussà et al. \(2020\)](#) curated a multilingual corpus of gender-balanced biographies from Wikipedia to facilitate unbiased research in machine translation. But this might not be

²³<https://medium.com/jigsaw/unintended-bias-and-names-of-frequently-targeted-groups-8e0b81f80a23>

possible from freely accessible and available data that models are mostly trained on. As a specific example, in one of our ongoing projects, we are attempting to collect task-specific data for indigenous languages. Gathering a representative dataset is immensely difficult especially owing to the very few language speakers accessible to us. Our conversations with more involved researchers working on indigenous Canadian languages affirmed that Simulating the data for underrepresented targets can be immensely helpful to other tasks in the NLP pipeline that have nothing to do with generation as well. Concretely, I experimented the effects of generating code-switched data from my approach described in §5.4 Chapter 5 to down-stream tasks. We observed an improvement by finetuning the model with our generated data before performing task-specific finetuning on various classification and sequence labeling tasks included in [Khanuja et al. \(2020\)](#). This problem is not limited to the diverse languages but variants of the same language as well. [Jurgens et al. \(2017\)](#) study the dialectal variability of English (which is unquestionably a resource-rich language) and studied how language identification tools are not socially equitable and fail in the dialectal variants of underdeveloped nations. Hence it is our responsibility to develop language technologies more equitably by accommodating all languages, dialects and controlling targeted text generation plays a pivotal role in generating task-specific datasets conducive to the labels. If few gold standard examples are present for the task or languages at hand, the generated datasets can be augmented with relatively low confidence samples for training.

Caution – Isolated anchoring or controlling: A root cause for not being able to ignore such imbalances is because models heavily learn from correlations but not from the causations. These correlations can have appalling effects on sensitive downstream applications such as AI-assisted therapists generating a life-ending suggestion. In large pre-trained models, this problem compounds with contextualized embeddings where these unintended correlations are propagated to the embeddings of other words as well ([Tan and Celis, 2019](#)). Furthermore, tackling the balance of datasets by generating anchored texts may inadvertently introduce other correlated biases. For instance, anchoring may remove biases against a class (say, women) and another class (say, black), but not against the aggregated class (say, black women). Therefore performing an independent and intersectional evaluation of the axes of biases is recommended before using this data for downstream purposes.

Faithfulness:

The gap between trusting the generation arises from misaligned incentives amongst the stakeholders: the researchers are inclined towards innovation working towards publications on research datasets, software companies tend to use the open-sourced technology off the shelf, adapting it to their use cases, and users consume this technology torn between the dilemma of trusting or not trusting the generated articles, etc., Improving the faithfulness of the generated text is of utmost importance to responsibly build the users' trust. Text generation models have leaped by gigantic means in generating sensible text from word salads. However, it is no hidden truth that it is easier for machines to write fiction than facts. In a world with rapid information exchange mushrooming fake news is detrimental to health in a pandemic ([de Barcelos et al., 2021](#)), political influences ([Van der Linden et al., 2020](#); [Hirst, 2017](#)), financial and stock markets ([Kogan et al., 2019](#); [Cheo, 2018](#)) etc., Faithful generation of text is, therefore, more crucial for almost all applications that can impact people and influence their notions. Potential ways to integrate this is by calibrating anchors' decoding with concepts identified from the images with

high confidence. This technique is used in decoding structure-to-text (Tian et al., 2019) with a confidence oriented decoder. Anchoring techniques are also used directly here: Wang et al. (2021b) use a two-staged skeleton-based method where the first stage is to carefully sketch skeleton generation with an autoregressive pointer network followed by non-autoregressive insertion and deletion operations to realize the surface form from the skeleton. This way of utilizing skeletons or anchors has been the central theme throughout the thesis. As discussed in §3.3.3.2 of Chapter 3, evaluating the prediction of anchors in the generated text is one form of interpretable and simplified testing strategies.

Caution – Personalization and Impersonation: Anchoring the generated text to a preferred personality can potentially be misused to generate believable texts like articles or emails impersonating a specific person. This can be used for social good like a celebrity promoting local businesses²⁴. However, we need to be attentive to the flip side of persona anchored generation that can impersonate an influential personality to spread fake information. For instance, multimodal disinformation (Hameleers et al., 2020; Alam et al., 2021) can range from harmless edits of restaurant dinner to harmful scenarios (Da et al., 2020). Learning to understand such edited media have extensive societal implications.

Explanation Generation:

The past few years witnessed vast strides in using language-based applications in everyday use that use at least a hint if not fully modeled by NLP techniques. These include critical domains such as healthcare, criminal justice, autonomous driving, etc.,. However, regulating these techniques and their usage is critical to bring it in as a way of our lives trust-worthily. Gade et al. (2020) present several such regulations like GDPR, Algorithmic Accountability Act imposed to make automated decision support systems accountable. Some of these guidelines include explaining the decisions of using this technology to the consumers²⁵. Generating explanations for individual cases is utmost useful for both the data scientists and the consumers to both trust this emerging technology and improve it when necessary. Amassing to other forms like visualizing model parameters and interpreting attention or interactive systems to improve predictions used by data scientists (Katsis and Wolf, 2019) which are popular, the text modality is a powerful means for generating human-understandable, natural language explanations. This conflates the advantages of explanation generation and explanation presentation into a single step. Goal-oriented narratives set a precedent for explanatory proofs. Generating explanations and rationalization is an analogy to a logical pathway towards arriving at proof. Certain tasks in NLP such as fake news detection, virality prediction, lie detection for courtroom judgments are better performed by machines than humans. Generating reasoning explanations for these scenarios can assist humans in learning patterns that machines can see and thus assist in making better and more informed judgments. Reasoning or explaining about the missing multimodal events requires understanding cross-modal information along with commonsense knowledge. In §3.4 of Chapter 3, I presented infilling in the inference in curriculum based learning model to bridge the missing information gap. Lei et al. (2020b) perform a similar task

²⁴<https://indianexpress.com/article/trending/trending-in-india/cadburys-diwali-shah-rukh-khan-7587613/>

²⁵<https://www.ftc.gov/news-events/blogs/business-blog/2020/04/using-artificial-intelligence-algorithms>

of reasoning about predicting the next event given a video and an associated dialog. The reasoning behind arriving at answer is also studied in the context of visual entailment (Do et al., 2020), visual question answering (Li et al., 2018b), visual commonsense reasoning (Zellers et al., 2019) etc., Similar to the principle used by Lattcinnik and Berant (2020) in the form of generating intermediate hypotheses using an LM to explain the answers predicted by a question answering model, predicting anchors derived from visual input to explain the downstream generation of multimodal stories serve as cross-modal explanations. This is the right direction pledged in progressing our field to bring this transformational yet rhapsodical technology closer to the hands of all the people without fear.

Caution – Misleading Explanations: Post-hoc explanation techniques for the black box predictions can be misleading due to several reasons like failure to capture causal relationships, and failure to be robust, falling prey to minor perturbations, etc., (Lakkaraju and Bastani, 2020). Understanding the sufficiency and comprehensiveness metrics (Carton et al., 2020) of an explanation can vary across fields and specific use cases. In such cases, it is crucial to work with domain experts to determine the standards of expected explanations that are useful in practice.

In the course of the thesis, I described how anchoring could tether and control narrative generation to various properties like content, structure, and surface forms from multiple views deriving from multiple modalities and languages. Based on these findings, I also described concrete promising future directions and a broader impact of the techniques proposed. These beneficial implications on various strata of the society encourage us to advance anchoring narrative generation from multiple views while carefully practicing caution to circumvent potentially intended or unintended ethical conundrums. With an optimistic outlook towards the future of multi-view anchored text generation, this thesis serves as a piece of Ariadne's string to navigate the labyrinth of potential that this field encompasses.

Bibliography

- Heike Adel, Ngoc Thang Vu, Katrin Kirchhoff, Dominic Telaar, and Tanja Schultz. 2015. Syntactic and semantic features for code-switching factored language models. *IEEE Transactions on Audio, Speech, and Language Processing*, 23(3):431–440.
- Heike Adel, Ngoc Thang Vu, Franziska Kraus, Tim Schlippe, Haizhou Li, and Tanja Schultz. 2013a. Recurrent neural network language modeling for code switching conversational speech. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8411–8415. IEEE.
- Heike Adel, Ngoc Thang Vu, and Tanja Schultz. 2013b. Combination of recurrent neural networks and factored language models for code-switching language modeling. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 206–211.
- Harsh Agrawal, Peter Anderson, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. [nocaps: novel object captioning at scale](#). In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 8947–8956. IEEE.
- Harsh Agrawal, Arjun Chandrasekaran, Dhruv Batra, Devi Parikh, and Mohit Bansal. 2016. Sort story: Sorting jumbled images and captions into stories. *arXiv preprint arXiv:1606.07493*.
- Arjun R. Akula, Spandana Gella, Yaser Al-Onaizan, Song-Chun Zhu, and Siva Reddy. 2020. [Words aren’t enough, their order matters: On the robustness of grounding visual referring expressions](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 6555–6565. Association for Computational Linguistics.
- Firoj Alam, Stefano Cresci, Tanmoy Chakraborty, Fabrizio Silvestri, Dimiter Dimitrov, Giovanni Da San Martino, Shaden Shaar, Hamed Firooz, and Preslav Nakov. 2021. [A survey on multimodal disinformation detection](#). *CoRR*, abs/2103.12541.
- Malihe Alikhani, Piyush Sharma, Shengjie Li, Radu Soricut, and Matthew Stone. 2020. Cross-modal coherence modeling for caption generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6525–6535.
- Afra Alishahi, Marie Barking, and Grzegorz Chrupala. 2017. [Encoding of phonology in a recurrent neural model of grounded speech](#). In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), Vancouver, Canada, August 3-4, 2017*, pages 368–378. Association for Computational Linguistics.

- Mehdi Allahyari, Seyedamin Pouriyeh, Mehdi Assefi, Saeid Safaei, Elizabeth D Trippe, Juan B Gutierrez, and Krys Kochut. 2017. Text summarization techniques: a brief survey. *arXiv preprint arXiv:1707.02268*.
- Prithviraj Ammanabrolu, William Broniec, Alex Mueller, Jeremy Paul, and Mark O Riedl. 2019a. Toward automated quest generation in text-adventure games. *arXiv preprint arXiv:1909.06283*.
- Prithviraj Ammanabrolu, Ethan Tien, Wesley Cheung, Zhaochen Luo, William Ma, Lara Martin, and Mark Riedl. 2019b. Guided neural language generation for automated storytelling. In *Proceedings of the Second Workshop on Storytelling*, pages 46–55.
- Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. [SPICE: semantic propositional image caption evaluation](#). In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V*, volume 9909 of *Lecture Notes in Computer Science*, pages 382–398. Springer.
- Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018a. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*.
- Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian D. Reid, Stephen Gould, and Anton van den Hengel. 2018b. [Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments](#). In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 3674–3683. Computer Vision Foundation / IEEE Computer Society.
- Jacob Andreas and Dan Klein. 2014. [Grounding language with points and paths in continuous spaces](#). In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning, CoNLL 2014, Baltimore, Maryland, USA, June 26-27, 2014*, pages 58–67. ACL.
- Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433.
- Sandhya Arora, Gauri Chaware, Devangi Chinchankar, Eesha Dixit, and Shevi Jain. 2019. Survey of different approaches used for food recognition. In *Information and Communication Technology for Competitive Strategies*, pages 551–560. Springer.
- Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.
- Ankit Arun, Soumya Batra, Vikas Bhardwaj, Ashwini Challa, Pinar Donmez, Peyman Heidari, Hakan Inan, Shashank Jain, Anuj Kumar, Shawn Mei, Karthik Mohan, and Michael White. 2020. [Best practices for data-efficient modeling in NLG: how to train production-ready neural models with less data](#). In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020 - Industry Track, Online, December 12, 2020*, pages 64–77. International Committee on Computational Linguistics.

- Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.
- Arun Baby, Nishanti NL, Anju Leela Thomas, and Heam A Myrthy. 2016. Resources for Indian Languages. In *Proceedings of Text, Speech and Dialogue*.
- Ashutosh Baheti, Alan Ritter, Jiwei Li, and Bill Dolan. 2018. [Generating more interesting responses in neural conversation models with distributional constraints](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 3970–3980. Association for Computational Linguistics.
- Anusha Balakrishnan, Jinfeng Rao, Kartikeya Upasani, Michael White, and Rajen Subba. 2019. [Constrained decoding for neural NLG from compositional representations in task-oriented dialogue](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 831–844. Association for Computational Linguistics.
- Satanjeev Banerjee and Alon Lavie. 2005a. [METEOR: an automatic metric for MT evaluation with improved correlation with human judgments](#). In *Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization@ACL 2005, Ann Arbor, Michigan, USA, June 29, 2005*, pages 65–72. Association for Computational Linguistics.
- Satanjeev Banerjee and Alon Lavie. 2005b. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.
- Hyunseung Bang, Antoni Virós Martin, Arnau Prat, and Daniel Selva. 2018. Daphne: An intelligent assistant for architecting earth observing satellite systems. In *2018 AIAA Information Systems-AIAA Infotech@ Aerospace*, page 1366.
- TDN de Barcelos, Luíza Nepomuceno Muniz, Deborah Marinho Dantas, Cotrim Junior DF, João Roberto Cavalcante, and Eduardo Faerstein. 2021. Analysis of fake news disseminated during the covid-19 pandemic in brazil. *Revista Panamericana de Salud Publica= Pan American Journal of Public Health*, 45:e65–e65.
- Loïc Barrault, Fethi Bougares, Lucia Specia, Chiraag Lala, Desmond Elliott, and Stella Frank. 2018. [Findings of the third shared task on multimodal machine translation](#). In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers, WMT 2018, Belgium, Brussels, October 31 - November 1, 2018*, pages 304–323. Association for Computational Linguistics.
- Lawrence W Barsalou, Karen Olseth Solomon, and Ling-Ling Wu. 1999. Perceptual simulation in conceptual tasks. *AMSTERDAM STUDIES IN THE THEORY AND HISTORY OF LINGUISTIC SCIENCE SERIES 4*, pages 209–228.
- Regina Barzilay and Noemie Elhadad. 2002. Inferring strategies for sentence ordering in multidocument news summarization. *Journal of Artificial Intelligence Research*, 17:35–55.

- Leonor Becerra-Bonache, Henning Christiansen, and M Dolores Jiménez-López. 2018. A gold standard to measure relative linguistic complexity with a grounded language learning model. In *Proceedings of the Workshop on Linguistic Complexity and Natural Language Processing*, pages 1–9.
- Andrew Begel, Daniel D. Garcia, and Steven A. Wolfman. 2004. [Kinesthetic learning in the classroom](#). In *Proceedings of the 35th SIGCSE Technical Symposium on Computer Science Education, SIGCSE 2004, Norfolk, Virginia, USA, March 3-7, 2004*, pages 183–184. ACM.
- Anja Belz. 2008. Automatic generation of weather forecast texts using comprehensive probabilistic generation-space models. *Natural Language Engineering*, 14(4):431–455.
- Anja Belz and Ehud Reiter. 2006. [Comparing automatic and human evaluation of NLG systems](#). In *EACL 2006, 11st Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference, April 3-7, 2006, Trento, Italy*. The Association for Computer Linguistics.
- Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. [Scheduled sampling for sequence prediction with recurrent neural networks](#). In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 1171–1179.
- Taylor Berg-Kirkpatrick, Dan Gillick, and Dan Klein. 2011. [Jointly learning to extract and compress](#). In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pages 481–490. The Association for Computer Linguistics.
- Ruth A Berman. 1988. On the ability to relate events in narrative. *Discourse Processes*, 11(4):469–497.
- Irshad Ahmad Bhat, Riyaz Ahmad Bhat, Manish Shrivastava, and Dipti Misra Sharma. 2018. Universal dependency parsing for hindi-english code-switching. *arXiv preprint arXiv:1804.05868*.
- Irshad Ahmad Bhat, Vandan Mujadia, Aniruddha Tammewar, Riyaz Ahmad Bhat, and Manish Shrivastava. 2014. Iiit-h system submission for fire2014 shared task on transliterated search. In *Proceedings of the Forum for Information Retrieval Evaluation*, pages 48–53. ACM.
- Yonatan Bisk, Jan Buys, Karl Pichotta, and Yejin Choi. 2019. [Benchmarking hierarchical script knowledge](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4077–4085. Association for Computational Linguistics.
- Yonatan Bisk, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, Nicolas Pinto, and Joseph P. Turian. 2020. [Experience grounds language](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 8718–8735. Association for Computational Linguistics.

- Yonatan Bisk, Siva Reddy, John Blitzer, Julia Hockenmaier, and Mark Steedman. 2016. [Evaluating induced CCG parsers on grounded semantic parsing](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 2022–2027. The Association for Computational Linguistics.
- Alan W Black. 2006. CLUSTERGEN: a statistical parametric synthesizer using trajectory modeling. In *INTERSPEECH*.
- Alan W Black, Kevin Lenzo, and Vincent Pagel. 1998. Issues in building general letter to sound rules. pages 77–80.
- Olivier Bodenreider. 2004. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270.
- Dan Bohus. Error awareness and recovery in task-oriented spoken dialogue systems.
- Tolga Bolukbasi, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and Adam Tauman Kalai. 2016. [Man is to computer programmer as woman is to homemaker? debiasing word embeddings](#). In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 4349–4357.
- Benjamin Börschinger, Bevan K. Jones, and Mark Johnson. 2011. [Reducing grounded learning tasks to grammatical inference](#). In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, 27-31 July 2011, John McIntyre Conference Centre, Edinburgh, UK, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1416–1425. ACL.
- Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. 2014. Food-101—mining discriminative components with random forests. In *European Conference on Computer Vision*, pages 446–461. Springer.
- Antoine Bosselut, Jianfu Chen, David Warren, Hannaneh Hajishirzi, and Yejin Choi. 2016. [Learning prototypical event structure from photo albums](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.
- Antoine Bosselut, Omer Levy, Ari Holtzman, Corin Ennis, Dieter Fox, and Yejin Choi. 2017. Simulating action dynamics with neural process networks. *arXiv preprint arXiv:1711.05313*.
- Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew M Dai, Rafal Jozefowicz, and Samy Bengio. 2015. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349*.
- Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew M. Dai, Rafal Józefowicz, and Samy Bengio. 2016. [Generating sentences from a continuous space](#). In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Berlin, Germany, August 11-12, 2016*, pages 10–21. ACL.
- William F. Brewer. 1982. [Plan understanding, narrative comprehension, and story schemas](#). In *Proceedings of the National Conference on Artificial Intelligence, Pittsburgh, PA, USA, August 18-20, 1982*, pages 262–264. AAAI Press.

- Peter F. Brown, Stephen Della Pietra, Vincent J. Della Pietra, Jennifer C. Lai, and Robert L. Mercer. 1992. An estimate of an upper bound for the entropy of english. *Comput. Linguistics*, 18(1):31–40.
- Jerome S Bruner. 1974. From communication to language—a psychological perspective. *Cognition*, 3(3):255–287.
- Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gasic. 2018. Multiwoz—a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026.
- Barbara E Bullock and Almeida Jacqueline Ed Toribio. 2009. *The Cambridge handbook of linguistic code-switching*. Cambridge University Press.
- Anaïs Cadilhac, Nicholas Asher, Farah Benamara, and Alex Lascarides. 2013. [Grounding strategic conversation: Using negotiation dialogues to predict trades in a win-lose game](#). In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 357–368. ACL.
- Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186.
- Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336. ACM.
- Ronald Cardenas, Ying Lin, Heng Ji, and Jonathan May. 2019. [A grounded unsupervised universal part-of-speech tagger for low-resource languages](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 2428–2439. Association for Computational Linguistics.
- Malinda Carpenter, Katherine Nagell, Michael Tomasello, George Butterworth, and Chris Moore. 1998. Social cognition, joint attention, and communicative competence from 9 to 15 months of age. *Monographs of the society for research in child development*, pages i–174.
- Samuel Carton, Anirudh Rathore, and Chenhao Tan. 2020. [Evaluating and characterizing human rationales](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 9294–9307. Association for Computational Linguistics.
- Isaac Caswell, Julia Kreutzer, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmungkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Javier Ortiz Suárez, Iroko Orife, Kelechi

- Ogueji, Rubungo Andre Niyongabo, Toan Q. Nguyen, Mathias Müller, André Müller, Shamsuddeen Hassan Muhammad, Nanda Muhammad, Ayanda Mnyakeni, Jamshidbek Mirzakhlov, Tapiwanashe Matangira, Colin Leong, Nze Lawson, Sneha Kudugunta, Yacine Jernite, Mathias Jenny, Orhan Firat, Bonaventure F. P. Dossou, Sakhile Dlamini, Nisansa de Silva, Sakine Çabuk Balli, Stella Biderman, Alessia Battisti, Ahmed Baruwa, Ankur Bapna, Pallavi Baljekar, Israel Abebe Azime, Ayodele Awokoya, Duygu Ataman, Orevaoghene Ahia, Oghenefego Ahia, Sweta Agrawal, and Mofetoluwa Adeyemi. 2021. [Quality at a glance: An audit of web-crawled multilingual datasets](#). *CoRR*, abs/2103.12028.
- Marc Cavazza, Fred Charles, and Steven J Mead. 2002. Interacting with virtual characters in interactive storytelling. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 1*, pages 318–325.
- Marc Cavazza, David Pizzi, Fred Charles, Thuri Vogt, and Elisabeth André. 2009. Emotional input for character-based interactive storytelling. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 313–320. International Foundation for Autonomous Agents and Multiagent Systems.
- Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2020. [Evaluation of text generation: A survey](#). *CoRR*, abs/2006.14799.
- Özlem Çetinoğlu and Çağrı Çöltekin. 2016. Part of speech annotation of a turkish-german code-switching corpus. In *Proceedings of the 10th Linguistic Annotation Workshop held in conjunction with ACL 2016 (LAW-X 2016)*, pages 120–130.
- Özlem Çetinoğlu, Sarah Schulz, and Ngoc Thang Vu. 2016. Challenges of computational processing of code-switching. *arXiv preprint arXiv:1610.02213*.
- Sunita Chand. 2016. Empirical survey of machine translation tools. In *2016 Second International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, pages 181–185. IEEE.
- Arjun Chandrasekaran, Devi Parikh, and Mohit Bansal. 2018. Punny captions: Witty wordplay in image descriptions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 770–775.
- Khyathi Chandu, Thomas Manzini, Sumeet Singh, and Alan W Black. 2018. Language informed modeling of code-switched text. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 92–97.
- Khyathi Chandu, Aakanksha Naik, Aditya Chandrasekar, Zi Yang, Niloy Gupta, and Eric Nyberg. 2017a. Tackling biomedical text summarization: Oaqa at bioasq 5b. *BioNLP 2017*, pages 58–66.
- Khyathi Chandu, Eric Nyberg, and Alan W Black. 2019a. Storyboarding of recipes: Grounded contextual generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6040–6046.

- Khyathi Chandu, Eric Nyberg, and Alan W Black. 2019b. Storyboarding of recipes: Grounded contextual generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6040–6046.
- Khyathi Chandu, Shrimai Prabhumoye, Ruslan Salakhutdinov, and Alan W Black. 2019c. “my way of telling a story”: Persona based grounded story generation. In *Proceedings of the Second Workshop on Storytelling*, pages 11–21.
- Khyathi Raghavi Chandu and Alan W Black. 2020a. Positioning yourself in the maze of neural text generation: A task-agnostic survey. *arXiv preprint arXiv:2010.07279*.
- Khyathi Raghavi Chandu and Alan W. Black. 2020b. [Style variation as a vantage point for code-switching](#). In *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*, pages 4761–4765. ISCA.
- Khyathi Raghavi Chandu, Ruo-Ping Dong, and Alan W. Black. 2020a. [Reading between the lines: Exploring infilling in visual narratives](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 1220–1229. Association for Computational Linguistics.
- Khyathi Raghavi Chandu, Sai Krishna Rallabandi, Sunayana Sitaram, and Alan W. Black. 2017b. [Speech synthesis for mixed-language navigation instructions](#). In *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, pages 57–61. ISCA.
- Khyathi Raghavi Chandu, Piyush Sharma, Soravit Changpinyo, Ashish Thapliyal, and Radu Soricut. 2020b. Denoising large-scale image captioning from alt-text data using content selection models. *arXiv preprint arXiv:2009.05175*.
- Angel X. Chang, Will Monroe, Manolis Savva, Christopher Potts, and Christopher D. Manning. 2015. [Text to 3d scene generation with rich lexical grounding](#). In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 53–62. The Association for Computer Linguistics.
- Chieh-Teng Chang, Chi-Chia Huang, and Jane Yung-jen Hsu. 2018a. [A hybrid word-character model for abstractive summarization](#). *CoRR*, abs/1802.09968.
- Ching-Ting Chang, Shun-Po Chuang, and Hung-Yi Lee. 2018b. Code-switching sentence generation by generative adversarial networks and its application to data augmentation. *arXiv preprint arXiv:1811.02356*.
- Chingching Chang. 2009. ” being hooked” by editorial content: The implications for processing narrative advertising. *Journal of Advertising*, 38(1):21–34.
- Soravit Changpinyo, Bo Pang, Piyush Sharma, and Radu Soricut. 2019. Decoupled box proposal and featurization with ultrafine-grained semantic labels improve image captioning and visual question answering. In *EMNLP-IJCNLP*.

- Snigdha Chaturvedi, Shashank Srivastava, and Dan Roth. 2018. [Where have I heard this story before? identifying narrative similarity in movie remakes](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 673–678. Association for Computational Linguistics.
- David L. Chen. 2012. [Fast online lexicon learning for grounded language acquisition](#). In *The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, July 8-14, 2012, Jeju Island, Korea - Volume 1: Long Papers*, pages 430–439. The Association for Computer Linguistics.
- Hongge Chen, Huan Zhang, Pin-Yu Chen, Jinfeng Yi, and Cho-Jui Hsieh. 2018. [Attacking visual language grounding with adversarial examples: A case study on neural image captioning](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2587–2597. Association for Computational Linguistics.
- Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. [A survey on dialogue systems: Recent advances and new frontiers](#). *SIGKDD Explorations*, 19(2):25–35.
- Mingda Chen, Qingming Tang, Sam Wiseman, and Kevin Gimpel. 2019a. [Controllable paraphrase generation with a syntactic exemplar](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 5972–5984. Association for Computational Linguistics.
- Mingda Chen, Qingming Tang, Sam Wiseman, and Kevin Gimpel. 2019b. A multi-task approach for disentangling syntax and semantics in sentence representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2453–2464.
- Ping Chen and Rakesh Verma. 2006. A query-based medical information summarization system using ontology knowledge. In *Computer-Based Medical Systems, 2006. CBMS 2006. 19th IEEE International Symposium on*, pages 37–42. IEEE.
- Qian Chen, Xiao-Dan Zhu, Zhen-Hua Ling, Si Wei, and Hui Jiang. 2016. [Distraction-based neural networks for modeling document](#). In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 2754–2760. IJCAI/AAAI Press.
- Shizhe Chen, Qin Jin, Peng Wang, and Qi Wu. 2020a. [Say as you wish: Fine-grained control of image caption generation with abstract scene graphs](#). *CoRR*, abs/2003.00387.
- Shizhe Chen, Qin Jin, Peng Wang, and Qi Wu. 2020b. [Say as you wish: Fine-grained control of image caption generation with abstract scene graphs](#). In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 9959–9968. IEEE.
- Yen-Chun Chen and Mohit Bansal. 2018. [Fast abstractive summarization with reinforcement selected sentence rewriting](#). In *Proceedings of the 56th Annual Meeting of the Association*

- for *Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 675–686. Association for Computational Linguistics.
- Yen-Chun Chen, Zhe Gan, Yu Cheng, Jingzhou Liu, and Jingjing Liu. 2019c. [Distilling the knowledge of BERT for text generation](#). *CoRR*, abs/1911.03829.
- Yen-Chun Chen, Zhe Gan, Yu Cheng, Jingzhou Liu, and Jingjing Liu. 2020c. [Distilling knowledge learned in BERT for text generation](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7893–7905. Association for Computational Linguistics.
- Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020d. UNITER: Learning UNiversal Image-TEText Representations. In *ECCV*.
- Zhenfang Chen, Lin Ma, Wenhan Luo, and Kwan-Yee Kenneth Wong. 2019d. [Weakly-supervised spatio-temporally grounding natural sentence in video](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1884–1894. Association for Computational Linguistics.
- Yu Cheng, Duo Wang, Pan Zhou, and Tao Zhang. 2017. A survey of model compression and acceleration for deep neural networks. *arXiv preprint arXiv:1710.09282*.
- James Cheo. 2018. Fake news can make–or break–stock prices. *The Business Times, The Business Times*, 5.
- Ta-Chung Chi, Minmin Shen, Mihail Eric, Seokhwan Kim, and Dilek Hakkani-Tür. 2020. [Just ask: An interactive learning framework for vision and language navigation](#). In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 2459–2466. AAAI Press.
- Russell KH Ching, Pingsheng Tong, Ja-Shen Chen, and Hung-Yen Chen. 2013. Narrative online advertising: identification and its effects on attitude toward a product. *Internet Research*.
- Hyundong Cho and Jonathan May. 2020. [Grounding conversations with improvised dialogues](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 2398–2413. Association for Computational Linguistics.
- Kyunghyun Cho. 2016. [Noisy parallel approximate decoding for conditional recurrent language model](#). *CoRR*, abs/1605.03835.
- Byung-Ju Choi, Jimin Hong, David Keetae Park, and Sang Wan Lee. 2020. [F²-softmax: Diversifying neural text generation via frequency factorized softmax](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 9167–9182. Association for Computational Linguistics.

- Sumit Chopra, Michael Auli, and Alexander M. Rush. 2016. [Abstractive sentence summarization with attentive recurrent neural networks](#). In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 93–98. The Association for Computational Linguistics.
- M Choudhury, G Chittaranjan, P Gupta, and A Das. 2014. Overview and datasets of fire 2014 track on transliterated search. In *Pre-proceedings 6th workshop FIRE-2014*.
- Grzegorz Chrupala. 2019. [Symbolic inductive bias for visually grounded learning of spoken language](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 6452–6462. Association for Computational Linguistics.
- Grzegorz Chrupala, Lieke Gelderloos, and Afra Alishahi. 2017. [Representations of language in a model of visually grounded speech signal](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 613–622. Association for Computational Linguistics.
- Chenhui Chu, Mayu Otani, and Yuta Nakashima. 2018. [iparaphrasing: Extracting visually grounded paraphrases via an image](#). In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 3479–3492. Association for Computational Linguistics.
- Chenhui Chu and Rui Wang. 2018. A survey of domain adaptation for neural machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1304–1319.
- Min Chu, Hu Peng, Yong Zhao, Zhengyu Niu, and Eric Chang. 2003. Microsoft Mulan-a bilingual TTS system. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, volume 1, pages I–I. IEEE.
- Junyoung Chung, Kyunghyun Cho, and Yoshua Bengio. 2016. A character-level decoder without explicit segmentation for neural machine translation. *arXiv preprint arXiv:1603.06147*.
- Elizabeth Clark, Yangfeng Ji, and Noah A. Smith. 2018a. [Neural text generation in stories using entity representations as context](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 2250–2260. Association for Computational Linguistics.
- Elizabeth Clark, Yangfeng Ji, and Noah A. Smith. 2018b. Neural text generation in stories using entity representations as context. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 2250–2260.
- Elizabeth Clark, Anne Spencer Ross, Chenhao Tan, Yangfeng Ji, and Noah A. Smith. 2018c. [Creative writing with a machine in the loop: Case studies on slogans and stories](#). In *Proceedings of the 23rd International Conference on Intelligent User Interfaces, IUI 2018, Tokyo, Japan, March 07-11, 2018*, pages 329–340. ACM.

- Elizabeth Clark and Noah A. Smith. 2021. [Choose your own adventure: Paired suggestions in collaborative writing for evaluating story generation models](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 3566–3575. Association for Computational Linguistics.
- Herbert H. Clark and Susan E. Brennan. 1991. [Grounding in communication](#). In Lauren B. Resnick, John M. Levine, and Stephanie D. Teasley, editors, *Perspectives on socially shared cognition*, pages 127–149. American Psychological Association.
- Herbert H Clark and Thomas B Carlson. 1982. Hearers and speech acts. *Language*, pages 332–373.
- Herbert H Clark and Meredyth A Krych. 2004. Speaking while monitoring addressees for understanding. *Journal of memory and language*, 50(1):62–81.
- Peter Clark. 2015. [Elementary school science and math tests as a driver for AI: take the aristo challenge!](#) In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, pages 4019–4021. AAAI Press.
- Peter Clark, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter D. Turney, and Daniel Khashabi. 2016. [Combining retrieval, statistics, and inference to answer elementary science questions](#). In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, pages 2580–2586. AAAI Press.
- Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, pages 7057–7067.
- Alexis Conneau, Guillaume Lample, Marc’Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word translation without parallel data. *arXiv preprint arXiv:1710.04087*.
- Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. 2019. Show, control and tell: A framework for generating controllable and grounded captions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8307–8316.
- Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. 2020. Meshed-Memory Transformer for Image Captioning. In *CVPR*.
- Marta R. Costa-jussà, Pau Li Lin, and Cristina España-Bonet. 2020. [Gebiotoolkit: Automatic extraction of gender-balanced multilingual corpus of wikipedia biographies](#). In *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*, pages 4081–4088. European Language Resources Association.
- Ryan Cotterell, Adithya Renduchintala, Naomi Saphra, and Chris Callison-Burch. 2014. An algerian arabic-french code-switched corpus. In *Workshop on Free/Open-Source Arabic Corpora and Corpora Processing Tools Workshop Programme*, page 34.
- Yin Cui, Guandao Yang, Andreas Veit, Xun Huang, and Serge J. Belongie. 2018. [Learning to evaluate image captioning](#). In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 5804–5812. IEEE Computer Society.

- Jeff Da, Maxwell Forbes, Rowan Zellers, Anthony Zheng, Jena D. Hwang, Antoine Bosselut, and Yejin Choi. 2020. [Edited media understanding: Reasoning about implications of manipulated images](#). *CoRR*, abs/2012.04726.
- Raj Dabre, Chenhui Chu, and Anoop Kunchukuttan. 2019. A survey of multilingual neural machine translation. *arXiv preprint arXiv:1905.05395*.
- Daniel Dahlmeier and Hwee Tou Ng. 2012. A beam-search decoder for grammatical error correction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 568–578. Association for Computational Linguistics.
- Bo Dai, Sanja Fidler, and Dahua Lin. 2018a. [A neural compositional paradigm for image captioning](#). In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 656–666.
- Bo Dai, Sanja Fidler, and Dahua Lin. 2018b. A neural compositional paradigm for image captioning. In *Advances in Neural Information Processing Systems*, pages 658–668.
- Robert Dale. 1992. *Generating referring expressions: Constructing descriptions in a domain of objects and processes*. The MIT Press.
- Robert Dale. 2006. Generating referring expressions.
- Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. 2018. [Embodied question answering](#). In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 1–10. Computer Vision Foundation / IEEE Computer Society.
- Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. 2017. Visual dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2.
- Amitava Das and Björn Gambäck. 2014. Identifying languages at the word level in code-mixed indian social media text.
- Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. [Plug and play language models: A simple approach to controlled text generation](#). In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.
- Hillary Dawkins. 2021. [Second order winobias \(sowinobias\) test set for latent gender bias detection in coreference resolution](#). *CoRR*, abs/2109.14047.
- Harm De Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron C Courville. 2017. Guesswhat?! visual object discovery through multi-modal dialogue. In *CVPR*, volume 1, page 3.
- Richard Delgado. 1989. Storytelling for oppositionists and others: A plea for narrative. *Michigan Law Review*, 87(8):2411–2441.

- Yuntian Deng and Alexander M. Rush. 2020. [Cascaded text generation with markov transformers](#). *CoRR*, abs/2006.01112.
- Margaret Deuchar, Peredur Davies, Jon Russell Herring, M Parafita Couto, and Diana Carter. 2014. Building bilingual corpora.
- Guy Deutscher. 2010. *Through the language glass: Why the world looks different in other languages*. Metropolitan Books.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019a. [BERT: pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019b. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.
- Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. [Wizard of wikipedia: Knowledge-powered conversational agents](#). In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.
- Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2018. [Measuring and mitigating unintended bias in text classification](#). In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2018, New Orleans, LA, USA, February 02-03, 2018*, pages 67–73. ACM.
- Virginie Do, Oana-Maria Camburu, Zeynep Akata, and Thomas Lukasiewicz. 2020. [e-snli-ve-2.0: Corrected visual-textual entailment with natural language explanations](#). *CoRR*, abs/2004.03744.
- George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the second international conference on Human Language Technology Research*, pages 138–145.
- Chris Donahue, Mina Lee, and Percy Liang. 2020. Enabling language models to fill in the blanks. *arXiv preprint arXiv:2005.05339*.
- Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019a. Unified language model pre-training for natural language understanding and generation. In *Advances in Neural Information Processing Systems*, pages 13042–13054.
- Ruo-Ping Dong, Khyathi Raghavi Chandu, and Alan W. Black. 2019b. [Induction and reference of entities in a visual story](#). *CoRR*, abs/1909.09699.

- Ruo-Ping Dong, Khyathi Raghavi Chandu, and Alan W Black. 2019c. Induction and reference of entities in a visual story. *arXiv preprint arXiv:1909.09699*.
- Kevin Donnelly. 2006. Snomed-ct: The advanced terminology and coding system for ehealth. *Studies in health technology and informatics*, 121:279.
- Gabriel Doyle and Michael C. Frank. 2015. [Shared common ground influences information density in microblog texts](#). In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 1587–1596. The Association for Computational Linguistics.
- John J Dudley, Keith Vertanen, and Per Ola Kristensson. 2018. Fast and precise touch-based text entry for head-mounted augmented reality with variable occlusion. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 25(6):30.
- Long Duong, Hadi Afshar, Dominique Estival, Glen Pink, Philip R Cohen, and Mark Johnson. 2017. Multilingual semantic parsing and code-switching. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 379–389.
- Nouha Dziri, Ehsan Kamaloo, Kory Mathewson, and Osmar R Zaiane. 2019. Augmenting neural response generation with context-aware topical attention. In *Proceedings of the First Workshop on NLP for Conversational AI*, pages 18–31.
- Abdessamad Echihabi and Daniel Marcu. 2003. A noisy-channel approach to question answering. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 16–23. Association for Computational Linguistics.
- Judith Eckle-Kohler. 2016. [Verbs taking clausal and non-finite arguments as signals of modality - revisiting the issue of meaning grounded in syntax](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.
- Desmond Elliott, Stella Frank, Khalil Sima'an, and Lucia Specia. 2016. [Multi30k: Multilingual english-german image descriptions](#). In *Proceedings of the 5th Workshop on Vision and Language, hosted by the 54th Annual Meeting of the Association for Computational Linguistics, August 12, Berlin, Germany*. The Association for Computer Linguistics.
- Desmond Elliott and Frank Keller. 2013. Image description using visual dependency representations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1292–1302.
- Naresh Kumar Elluru, Anandaswarup Vadapalli, Raghavendra Elluru, Hema Murthy, and Kishore Prahallad. 2013. Is word-to-phone mapping better than phone-phone mapping for handling english words? In *ACL (2)*, pages 196–200.
- Michael Emonts, Rebecca Row, W Lewis Johnson, Elizabeth Thomson, Helen de Silva Joyce, LTCOL Giles Gorman, and Robert Carpenter. 2012. Integration of social simulations into a task-based blended training curriculum. In *Land warfare conference, Melbourne, Australia*.

- Matthew E Falagas, Eleni I Pitsouni, George A Malietzis, and Georgios Pappas. 2008. Comparison of pubmed, scopus, web of science, and google scholar: strengths and weaknesses. *The FASEB journal*, 22(2):338–342.
- Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898.
- Angela Fan, Mike Lewis, and Yann Dauphin. 2019a. Strategies for structuring story generation. *arXiv preprint arXiv:1902.01109*.
- Angela Fan, Mike Lewis, and Yann N. Dauphin. 2019b. [Strategies for structuring story generation](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 2650–2660. Association for Computational Linguistics.
- Zhihao Fan, Zhongyu Wei, Siyuan Wang, and Xuanjing Huang. 2019c. [Bridging by word: Image grounded vocabulary construction for visual captioning](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 6514–6524. Association for Computational Linguistics.
- Hao Fang, Saurabh Gupta, Forrest N. Iandola, Rupesh Kumar Srivastava, Li Deng, Piotr Dollár, Jianfeng Gao, Xiaodong He, Margaret Mitchell, John C. Platt, C. Lawrence Zitnick, and Geoffrey Zweig. 2015. [From captions to visual concepts and back](#). In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 1473–1482. IEEE Computer Society.
- William Fedus, Ian J. Goodfellow, and Andrew M. Dai. 2018a. [Maskgan: Better text generation via filling in the _____](#). In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.
- William Fedus, Ian J. Goodfellow, and Andrew M. Dai. 2018b. [Maskgan: Better text generation via filling in the _____](#). In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.
- Yang Feng, Lin Ma, Wei Liu, and Jiebo Luo. 2019. [Unsupervised image captioning](#). In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 4125–4134. Computer Vision Foundation / IEEE.
- Oliver Ferschke, Johannes Daxenberger, and Iryna Gurevych. 2013. [A survey of NLP methods and resources for analyzing the collaborative writing process in wikipedia](#). In Iryna Gurevych and Jungi Kim, editors, *The People’s Web Meets NLP, Collaboratively Constructed Language Resources*, Theory and Applications of Natural Language Processing, pages 121–160. Springer.
- Katja Filippova. 2010. Multi-sentence compression: Finding shortest paths in word graphs. In *In Proceedings of the 23rd International Conference on Computational Linguistics*, pages 322–330. ACL.

- Katja Filippova, Enrique Alfonseca, Carlos A Colmenares, Lukasz Kaiser, and Oriol Vinyals. 2015. Sentence compression by deletion with lstms. In *EMNLP*, pages 360–368.
- Michael Fleischman and Deb Roy. 2008. [Grounded language modeling for automatic speech recognition of sports video](#). In *ACL 2008, Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics, June 15-20, 2008, Columbus, Ohio, USA*, pages 121–129. The Association for Computer Linguistics.
- Maxwell Forbes, Jena D. Hwang, Vered Shwartz, Maarten Sap, and Yejin Choi. 2020. [Social chemistry 101: Learning to reason about social and moral norms](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 653–670. Association for Computational Linguistics.
- Markus Freitag, David Grangier, and Isaac Caswell. 2020. [BLEU might be guilty but references are not innocent](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 61–71. Association for Computational Linguistics.
- Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2018. Style transfer in text: Exploration and evaluation. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Christian Fügen, Alex Waibel, and Muntsin Kolss. 2007. Simultaneous translation of lectures and speeches. *Machine translation*, 21(4):209–252.
- Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. 2016. [Multimodal compact bilinear pooling for visual question answering and visual grounding](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 457–468. The Association for Computational Linguistics.
- Saadia Gabriel, Asli Çelikyilmaz, Rahul Jha, Yejin Choi, and Jianfeng Gao. 2020. [Go figure! A meta evaluation of factuality in summarization](#). *CoRR*, abs/2010.12834.
- Krishna Gade, Sahin Cem Geyik, Krishnaram Kenthapadi, Varun Mithal, and Ankur Taly. 2020. [Explainable AI in industry: Practical challenges and lessons learned](#). In *Companion of the 2020 Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, pages 303–304. ACM / IW3C2.
- Chuang Gan, Zhe Gan, Xiaodong He, Jianfeng Gao, and Li Deng. 2017. [Stylenet: Generating attractive visual captions with styles](#). In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 955–964. IEEE Computer Society.
- Qiaozi Gao, Malcolm Doering, Shaohua Yang, and Joyce Yue Chai. 2016. [Physical causality of action verbs in grounded language understanding](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.
- David García, Ingmar Weber, and Venkata Rama Kiran Garimella. 2014. [Gender asymmetries in reality and fiction: The bechdel test of social media](#). In *Proceedings of the Eighth International Conference on Weblogs and Social Media, ICWSM 2014, Ann Arbor, Michigan, USA, June 1-4, 2014*. The AAAI Press.

- Norberto Fernández García, Luis Sánchez Fernández, José M. Blázquez del Toro, and Jesús Villamor-Lugo. 2007. [The news ontology for professional journalism applications](#). In Raj Sharman, Rajiv Kishore, and Ram Ramesh, editors, *Ontologies: A Handbook of Principles, Concepts and Applications in Information Systems*, volume 14 of *Integrated Series in Information Systems*, pages 887–919. Springer.
- Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. [The webnlg challenge: Generating text from RDF data](#). In *Proceedings of the 10th International Conference on Natural Language Generation, INLG 2017, Santiago de Compostela, Spain, September 4-7, 2017*, pages 124–133. Association for Computational Linguistics.
- Nikhil Garg, Londa Schiebinger, Dan Jurafsky, and James Zou. 2018a. [Word embeddings quantify 100 years of gender and ethnic stereotypes](#). *Proc. Natl. Acad. Sci. USA*, 115(16):E3635–E3644.
- Saurabh Garg, Tanmay Parekh, and Preethi Jyothi. 2017. Dual language models for code mixed speech recognition. *arXiv preprint arXiv:1711.01048*.
- Saurabh Garg, Tanmay Parekh, and Preethi Jyothi. 2018b. Code-switched language models using dual rnns and same-source pretraining. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3078–3083.
- Albert Gatt and Emiel Krahmer. 2018a. [Survey of the state of the art in natural language generation: Core tasks, applications and evaluation](#). *J. Artif. Intell. Res.*, 61:65–170.
- Albert Gatt and Emiel Krahmer. 2018b. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research*, 61:65–170.
- Sebastian Gehrmann, Tosin P. Adewumi, Karmanya Aggarwal, Pawan Sasanka Ammanamanchi, Aremu Anuoluwapo, Antoine Bosselut, Khyathi Raghavi Chandu, Miruna-Adriana Clinciu, Dipanjan Das, Kaustubh D. Dhole, Wanyu Du, Esin Durmus, Ondrej Dusek, Chris Emezue, Varun Gangal, Cristina Garbacea, Tatsunori Hashimoto, Yufang Hou, Yacine Jernite, Harsh Jhamtani, Yangfeng Ji, Shailza Jolly, Dhruv Kumar, Faisal Ladhak, Aman Madaan, Mounica Maddela, Khyati Mahajan, Saad Mahamood, Bodhisattwa Prasad Majumder, Pedro Henrique Martins, Angelina McMillan-Major, Simon Mille, Emiel van Miltenburg, Moin Nadeem, Shashi Narayan, Vitaly Nikolaev, Rubungo Andre Niyongabo, Salomey Osei, Ankur P. Parikh, Laura Perez-Beltrachini, Niranjan Ramesh Rao, Vikas Raunak, Juan Diego Rodriguez, Sashank Santhanam, João Sedoc, Thibault Sellam, Samira Shaikh, Anastasia Shimorina, Marco Antonio Sobrevilla Cabezudo, Hendrik Strobelt, Nishant Subramani, Wei Xu, Diyi Yang, Akhila Yerukola, and Jiawei Zhou. 2021. [The GEM benchmark: Natural language generation, its evaluation and metrics](#). *CoRR*, abs/2102.01672.
- Sebastian Gehrmann, Yuntian Deng, and Alexander M Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4098–4109.
- Spandana Gella, Mike Lewis, and Marcus Rohrbach. 2018. A dataset for telling the stories of social media videos. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 968–974.

- Pablo Gervás, Belén Díaz-Agudo, Federico Peinado, and Raquel Hervás. 2004. Story plot generation based on cbr. In *International Conference on Innovative Techniques and Applications of Artificial Intelligence*, pages 33–46. Springer.
- Mor Geva, Yoav Goldberg, and Jonathan Berant. 2019. [Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 1161–1166. Association for Computational Linguistics.
- Marjan Ghazvininejad, Chris Brockett, Ming-Wei Chang, Bill Dolan, Jianfeng Gao, Wen-tau Yih, and Michel Galley. 2018. [A knowledge-grounded neural conversation model](#). In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5110–5117. AAAI Press.
- Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. 2019. [Mask-predict: Parallel decoding of conditional masked language models](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 6111–6120. Association for Computational Linguistics.
- Marjan Ghazvininejad, Omer Levy, and Luke Zettlemoyer. 2020. Semi-autoregressive training improves mask-predict decoding. *arXiv preprint arXiv:2001.08785*.
- Nabeel Gillani, Taha Yasseri, Rebecca Eynon, and Isis Hjorth. 2014. [Structural limitations of learning in a crowd: communication vulnerability and information diffusion in moocs](#). *CoRR*, abs/1411.3662.
- Dan Gillick and Benoit Favre. 2009. A scalable global model for summarization. In *In Proceedings of NAACL*.
- Daniel Gillick, Benoît Favre, and Dilek Hakkani-Tür. 2008. [The ICSI summarization system at TAC 2008](#). In *Proceedings of the First Text Analysis Conference, TAC 2008, Gaithersburg, Maryland, USA, November 17-19, 2008*. NIST.
- Goran Glavas and Sanja Stajner. 2015. [Simplifying lexical simplification: Do we need simplified corpora?](#) In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 2: Short Papers*, pages 63–68. The Association for Computer Linguistics.
- Ruediger Glott, Philipp Schmidt, and Rishab Ghosh. 2010. Wikipedia survey—overview of results. *United Nations University: Collaborative Creativity Group*, 8:1158–1178.
- Hila Gonen and Yoav Goldberg. 2019. [Lipstick on a pig: Debiasing methods cover up systematic gender biases in word embeddings but do not remove them](#). In *Proceedings of the 2019 Workshop on Widening NLP@ACL 2019, Florence, Italy, July 28, 2019*, pages 60–63. Association for Computational Linguistics.

- Diana Gonzalez-Rico and Gibran Fuentes-Pineda. 2018. Contextualize, show and tell: a neural visual storyteller. *arXiv preprint arXiv:1806.00738*.
- Xavi Gonzalvo, Siamak Tazari, Chun-an Chan, Markus Becker, Alexander Gutkin, and Hanna Silen. 2016. Recent advances in google real-time hmm-driven unit selection synthesizer.
- P Goyal, Manav R Mital, A Mukerjee, Achla M Raina, D Sharma, P Shukla, and K Vikram. 2003. A bilingual parser for hindi, english and code-switching structures. In *10th Conference of The European Chapter*, page 15.
- Tanya Goyal and Greg Durrett. 2020. [Evaluating factuality in generation with dependency-level entailment](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020, Online Event, 16-20 November 2020*, pages 3592–3603. Association for Computational Linguistics.
- David Graff, Shudong Huang, Ingrid Cartagena, Kevin Walker, and Christopher Cieri. 2010. Fisher spanish transcripts ldc2010t04. *Web Download, Philadelphia, USA*.
- Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, et al. 2016. Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626):471–476.
- Robert Gray. 1984. Vector quantization. *IEEE Assp Magazine*, 1(2):4–29.
- Anthony G Greenwald, Debbie E McGhee, and Jordan LK Schwartz. 1998. Measuring individual differences in implicit cognition: the implicit association test. *Journal of personality and social psychology*, 74(6):1464.
- Alvin Grissom II, He He, Jordan Boyd-Graber, John Morgan, and Hal Daumé III. 2014. Don’t until the final verb wait: Reinforcement learning for simultaneous machine translation. In *Proceedings of the 2014 Conference on empirical methods in natural language processing (EMNLP)*, pages 1342–1352.
- Barbara J Grosz, Scott Weinstein, and Aravind K Joshi. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational linguistics*, 21(2):203–225.
- Aditya Grover, Jiaming Song, Ashish Kapoor, Kenneth Tran, Alekh Agarwal, Eric Horvitz, and Stefano Ermon. 2019. [Bias correction of learned generative models using likelihood-free importance weighting](#). In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 11056–11068.
- Dusan Grujicic, Gorjan Radevski, Tinne Tuytelaars, and Matthew B. Blaschko. 2020. [Learning to ground medical text in a 3d human atlas](#). In *Proceedings of the 24th Conference on Computational Natural Language Learning, CoNLL 2020, Online, November 19-20, 2020*, pages 302–312. Association for Computational Linguistics.
- Jiatao Gu, Daniel Jiwoong Im, and Victor OK Li. 2018a. Neural machine translation with gumbel-greedy decoding. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

- Jiatao Gu, Qi Liu, and Kyunghyun Cho. 2019a. [Insertion-based decoding with automatically inferred generation order](#). *Trans. Assoc. Comput. Linguistics*, 7:661–676.
- Jiatao Gu, Zhengdong Lu, Hang Li, and Victor OK Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1631–1640.
- Jiatao Gu, Graham Neubig, Kyunghyun Cho, and Victor O. K. Li. 2017. [Learning to translate in real-time with neural machine translation](#). In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 1: Long Papers*, pages 1053–1062. Association for Computational Linguistics.
- Jiuxiang Gu, Shafiq Joty, Jianfei Cai, and Gang Wang. 2018b. Unpaired image captioning by language pivoting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 503–519.
- Jiuxiang Gu, Shafiq R. Joty, Jianfei Cai, and Gang Wang. 2018c. [Unpaired image captioning by language pivoting](#). In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part I*, volume 11205 of *Lecture Notes in Computer Science*, pages 519–535. Springer.
- Jiuxiang Gu, Shafiq R. Joty, Jianfei Cai, Handong Zhao, Xu Yang, and Gang Wang. 2019b. [Unpaired image captioning via scene graph alignments](#). In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 10322–10331. IEEE.
- Caglar Gulcehre, Sungjin Ahn, Ramesh Nallapati, Bowen Zhou, and Yoshua Bengio. 2016. Pointing the unknown words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 140–149.
- Jiaxian Guo, Sidi Lu, Han Cai, Weinan Zhang, Yong Yu, and Jun Wang. 2018. [Long text generation via adversarial training with leaked information](#). In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5141–5148. AAAI Press.
- Junliang Guo, Xu Tan, Di He, Tao Qin, Linli Xu, and Tie-Yan Liu. 2019a. [Non-autoregressive neural machine translation with enhanced decoder input](#). In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 3723–3730. AAAI Press.
- Junliang Guo, Xu Tan, Linli Xu, Tao Qin, Enhong Chen, and Tie-Yan Liu. 2020. [Fine-tuning by curriculum learning for non-autoregressive neural machine translation](#). In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on*

- Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7839–7846. AAAI Press.
- Longteng Guo, Jing Liu, Peng Yao, Jiangwei Li, and Hanqing Lu. 2019b. Mscap: Multi-style image captioning with unpaired stylized text. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4204–4213.
- Gualberto A Guzmán, Joseph Ricard, Jacqueline Serigos, Barbara E Bullock, and Almeida Jacqueline Toribio. 2017. Metrics for modeling code-switching across corpora.
- Janosch Haber, Tim Baumgärtner, Ece Takmaz, Lieke Gelderloos, Elia Bruni, and Raquel Fernández. 2019. [The photobook dataset: Building common ground through visually-grounded dialogue](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1895–1910. Association for Computational Linguistics.
- Michael Hameleers, Thomas E Powell, Toni GLA Van Der Meer, and Lieke Bos. 2020. A picture paints a thousand lies? the effects and mechanisms of multimodal disinformation and rebuttals disseminated via social media. *Political Communication*, 37(2):281–301.
- Ting Han and David Schlangen. 2017. [Grounding language by continuous observation of instruction following](#). In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers*, pages 491–496. Association for Computational Linguistics.
- Brent Harrison and Mark O. Riedl. 2016. [Learning from stories: Using crowdsourced narratives to train virtual agents](#). In *Proceedings of the Twelfth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, AIIDE 2016, October 8-12, 2016, Burlingame, California, USA*, pages 183–189. AAAI Press.
- Vrindavan Harrison, Lena Reed, Shereen Oraby, and Marilyn Walker. 2019. Maximizing stylistic control and semantic accuracy in nlg: Personality variation and discourse contrast. *DSNNG 2019*, page 1.
- Tatsunori B. Hashimoto, Hugh Zhang, and Percy Liang. 2019. [Unifying human and statistical evaluation for natural language generation](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 1689–1701. Association for Computational Linguistics.
- Dawnene D Hassett and Jen Scott Curwood. 2009. Theories and practices of multimodal education: The instructional dynamics of picture books and primary classrooms. *The Reading Teacher*, 63(4):270–282.
- William Havard, Laurent Besacier, and Jean-Pierre Chevrot. 2020. [Catplayinginthesnow: Impact of prior segmentation on a model of visually grounded speech](#). In *Proceedings of the 24th Conference on Computational Natural Language Learning, CoNLL 2020, Online, November 19-20, 2020*, pages 291–301. Association for Computational Linguistics.

- William N. Havard, Jean-Pierre Chevrot, and Laurent Besacier. 2019. [Word recognition, competition, and activation in a model of visually grounded speech](#). In *Proceedings of the 23rd Conference on Computational Natural Language Learning, CoNLL 2019, Hong Kong, China, November 3-4, 2019*, pages 339–348. Association for Computational Linguistics.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016a. Deep residual learning for image recognition. In *CVPR*.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016b. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.
- Matthew Henderson, Blaise Thomson, and Jason D Williams. 2014. The second dialog state tracking challenge. In *Proceedings of the 15th annual meeting of the special interest group on discourse and dialogue (SIGDIAL)*, pages 263–272.
- Jack Hessel, Zhenhai Zhu, Bo Pang, and Radu Soricut. 2020. [Beyond instructional videos: Probing for more diverse visual-textual grounding on youtube](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 8812–8822. Association for Computational Linguistics.
- Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. In *Proceedings of NIPS workshop*.
- Martin Hirst. 2017. Towards a political economy of fake news. *The political economy of communication*, 5(2).
- Aleksandra Hollingshead. 2018. Designing engaging online environments: Universal design for learning principles. In *Cultivating diverse online classrooms through effective instructional design*, pages 280–298. IGI Global.
- Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. [The curious case of neural text degeneration](#). In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.
- Ari Holtzman, Jan Buys, Maxwell Forbes, Antoine Bosselut, David Golub, and Yejin Choi. 2018. [Learning to write with cooperative discriminators](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 1638–1649. Association for Computational Linguistics.
- Mahmud Shahriar Hossain, Patrick Butler, Arnold P. Boedihardjo, and Naren Ramakrishnan. 2012. [Storytelling in entity networks to support intelligence analysts](#). In *The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '12, Beijing, China, August 12-16, 2012*, pages 1375–1383. ACM.
- MD Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. 2019. A comprehensive survey of deep learning for image captioning. *ACM Computing Surveys (CSUR)*, 51(6):118.
- Md. Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. 2018. [A comprehensive survey of deep learning for image captioning](#). *CoRR*, abs/1810.04020.

- Jingyi Hou, Xinxiao Wu, Xiaoxun Zhang, Yayun Qi, Yunde Jia, and Jiebo Luo. 2020. [Joint commonsense and relation reasoning for image and video captioning](#). In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 10973–10980. AAAI Press.
- Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339.
- David M. Howcroft, Anya Belz, Miruna-Adriana Clinciu, Dimitra Gkatzia, Sadid A Hasan, Saad Mahamood, Simon Mille, Emiel van Miltenburg, Sashank Santhanam, and Verena Rieser. 2020. [Twenty years of confusion in human evaluation: NLG needs evaluation sheets and standardised definitions](#). In *Proceedings of the 13th International Conference on Natural Language Generation, INLG 2020, Dublin, Ireland, December 15-18, 2020*, pages 169–182. Association for Computational Linguistics.
- Chao-Chun Hsu, Szu-Min Chen, Ming-Hsun Hsieh, and Lun-Wei Ku. 2018. Using inter-sentence diverse beam search to reduce redundancy in visual storytelling. *arXiv preprint arXiv:1805.11867*.
- Junjie Hu, Yu Cheng, Zhe Gan, Jingjing Liu, Jianfeng Gao, and Graham Neubig. 2020a. [What makes A good story? designing composite rewards for visual storytelling](#). In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7969–7976. AAAI Press.
- Junjie Hu, Yu Cheng, Zhe Gan, Jingjing Liu, Jianfeng Gao, and Graham Neubig. 2020b. [What makes A good story? designing composite rewards for visual storytelling](#). In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7969–7976. AAAI Press.
- Ronghang Hu, Daniel Fried, Anna Rohrbach, Dan Klein, Trevor Darrell, and Kate Saenko. 2019. [Are you looking? grounding to multiple modalities in vision-and-language navigation](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 6551–6557. Association for Computational Linguistics.
- Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1587–1596. JMLR. org.
- Lun Huang, Wenmin Wang, Jie Chen, and Xiao-Yong Wei. 2019a. Attention on attention for image captioning. In *CVPR*.

- Pingping Huang, Jianhui Huang, Yuqing Guo, Min Qiao, and Yong Zhu. 2019b. [Multi-grained attention with object-level grounding for visual question answering](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3595–3600. Association for Computational Linguistics.
- Poyao Huang, Mandela Patrick, Junjie Hu, Graham Neubig, Florian Metze, and Alex Hauptmann. 2021. [Multilingual multimodal pre-training for zero-shot cross-lingual transfer of vision-language models](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 2443–2459. Association for Computational Linguistics.
- Qiuyuan Huang, Zhe Gan, Asli Celikyilmaz, Dapeng Wu, Jianfeng Wang, and Xiaodong He. 2019c. Hierarchically structured reinforcement learning for topically coherent visual story generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8465–8472.
- Ting-Hao Kenneth Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Aishwarya Agrawal, Jacob Devlin, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, et al. 2016. Visual storytelling. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1233–1239.
- Radhwan Hussein Ibrahim and Dhia-Alrahman Hussein. 2016. Assessment of visual, auditory, and kinesthetic learning style among undergraduate nursing students. *Int J Adv Nurs Stud*, 5(1):1–4.
- Gabriel Ilharco, Yuan Zhang, and Jason Baldridge. 2019. [Large-scale representation learning from visually grounded untranscribed speech](#). In *Proceedings of the 23rd Conference on Computational Natural Language Learning, CoNLL 2019, Hong Kong, China, November 3-4, 2019*, pages 55–65. Association for Computational Linguistics.
- Daphne Ippolito, David Grangier, Chris Callison-Burch, and Douglas Eck. 2019. Unsupervised hierarchical story infilling. In *Proceedings of the First Workshop on Narrative Understanding*, pages 37–43.
- Touseef Iqbal and Shaima Qureshi. 2020. The survey: Text generation models in deep learning. *Journal of King Saud University-Computer and Information Sciences*.
- Mohit Iyyer, Varun Manjunatha, Anupam Guha, Yogarshi Vyas, Jordan L. Boyd-Graber, Hal Daumé III, and Larry S. Davis. 2017. [The amazing mysteries of the gutter: Drawing inferences between panels in comic book narratives](#). In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 6478–6487. IEEE Computer Society.
- Sarthak Jain and Byron C Wallace. 2019. Attention is not explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3543–3556.

- Anupam Jamatia, Björn Gambäck, and Amitava Das. 2015. Part-of-speech tagging for code-mixed english-hindi twitter and facebook chat messages. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 239–248.
- Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.
- Michaela Jänner, Karthik Narasimhan, and Regina Barzilay. 2018. [Representation learning for grounded spatial reasoning](#). *Trans. Assoc. Comput. Linguistics*, 6:49–61.
- Sujay Kumar Jauhar, Chris Dyer, and Eduard H. Hovy. 2015. [Ontologically grounded multi-sense representation learning for semantic vector space models](#). In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 683–693. The Association for Computational Linguistics.
- Sen Jia, Thomas Lansdall-Welfare, Saatviga Sudhahar, Cynthia Carter, and Nello Cristianini. 2016. Women are seen more than heard in online newspapers. *PloS one*, 11(2):e0148434.
- Xin Jia, Wenjie Zhou, Xu Sun, and Yunfang Wu. 2021. [EQG-RACE: examination-type question generation](#). In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13143–13151. AAAI Press.
- Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2020a. [How can we know when language models know?](#) *CoRR*, abs/2012.00955.
- Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020b. [How can we know what language models know](#). *Trans. Assoc. Comput. Linguistics*, 8:423–438.
- Xisen Jin, Junyi Du, Arka Sadhu, Ram Nevatia, and Xiang Ren. 2020. [Visually grounded continual learning of compositional phrases](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 2018–2029. Association for Computational Linguistics.
- Vineet John, Lili Mou, Hareesh Bahuleyan, and Olga Vechtomova. 2019. Disentangled representation learning for non-parallel text style transfer. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 424–434.
- Justin Johnson, Andrej Karpathy, and Li Fei-Fei. 2016. [Densecap: Fully convolutional localization networks for dense captioning](#). In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 4565–4574. IEEE Computer Society.
- Mark Johnson, Katherine Demuth, and Michael C. Frank. 2012. [Exploiting social information in grounded language learning via grammatical reduction](#). In *The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, July 8-14, 2012, Jeju Island, Korea - Volume 1: Long Papers*, pages 883–891. The Association for Computer Linguistics.

- W Lewis Johnson and Sara Behani Zaker. 2012. The power of social simulation for chinese language teaching.
- John Jonides. 1981. Voluntary versus automatic control over the mind's eye's movement. *Attention and performance IX*, 9:187–203.
- Aravind K Joshi. 1982. Processing of sentences with intra-sentential code-switching. In *Proceedings of the 9th conference on Computational linguistics-Volume 1*, pages 145–150. Academia Praha.
- Da-Cheng Juan, Chun-Ta Lu, Zhen Li, Futang Peng, Aleksei Timofeev, Yi-Ting Chen, Yaxi Gao, Tom Duerig, Andrew Tomkins, and Sujith Ravi. 2019. [Graph-rise: Graph-regularized image semantic embedding](#). *CoRR*, abs/1902.10814.
- Da-Cheng Juan, Chun-Ta Lu, Zhen Li, Futang Peng, Aleksei Timofeev, Yi-Ting Chen, Yaxi Gao, Tom Duerig, Andrew Tomkins, and Sujith Ravi. 2020. Ultra fine-grained image semantic embedding. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, pages 277–285.
- Yunjae Jung, Dahun Kim, Sanghyun Woo, Kyungsu Kim, Sungjin Kim, and In So Kweon. 2020. [Hide-and-tell: Learning to bridge photo streams for visual storytelling](#). In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 11213–11220. AAAI Press.
- David Jurgens, Yulia Tsvetkov, and Dan Jurafsky. 2017. [Incorporating dialectal variability for socially equitable language identification](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pages 51–57. Association for Computational Linguistics.
- Hiroataka Kameko, Shinsuke Mori, and Yoshimasa Tsuruoka. 2015. [Can symbol grounding improve low-level nlp? word segmentation as a case study](#). In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 2298–2303. The Association for Computational Linguistics.
- Daniel Kang and Tatsunori Hashimoto. 2020. [Improved natural language generation via loss truncation](#). *CoRR*, abs/2004.14589.
- Yannis Katsis and Christine T. Wolf. 2019. [Modellens: An interactive system to support the model improvement practices of data science teams](#). In *Companion Publication of the 2019 ACM Conference on Computer Supported Cooperative Work and Social Computing, CSCW 2019, Austin, TX, USA, November 09-13, 2019*, pages 9–13. ACM.
- Kazuya Kawakami, Chris Dyer, and Phil Blunsom. 2019. [Learning to discover, ground and use words with segmental neural language models](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 6429–6441. Association for Computational Linguistics.

- John D. Kelleher, Geert-Jan M. Kruijff, and Fintan J. Costello. 2006. [Proximity in context: An empirically grounded computational model of proximity for processing topological spatial expressions](#). In *ACL 2006, 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, Sydney, Australia, 17-21 July 2006*. The Association for Computer Linguistics.
- Casey Kennington and David Schlangen. 2015. [Simple learning and compositional application of perceptually grounded word meanings for incremental reference resolution](#). In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 292–301. The Association for Computer Linguistics.
- Ahmed Khalifa, Gabriella AB Barros, and Julian Togelius. 2017. Deepingle. *arXiv preprint arXiv:1705.03557*.
- Simran Khanuja, Sandipan Dandapat, Anirudh Srinivasan, Sunayana Sitaram, and Monojit Choudhury. 2020. [Gluecos: An evaluation benchmark for code-switched NLP](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 3575–3585. Association for Computational Linguistics.
- Chloé Kiddon, Luke Zettlemoyer, and Yejin Choi. 2016. Globally coherent text generation with neural checklist models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 329–339.
- Douwe Kiela, Luana Bulat, and Stephen Clark. 2015. [Grounding semantics in olfactory perception](#). In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 2: Short Papers*, pages 231–236. The Association for Computer Linguistics.
- Douwe Kiela and Stephen Clark. 2015. [Multi- and cross-modal semantics beyond vision: Grounding in auditory perception](#). In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 2461–2470. The Association for Computational Linguistics.
- Mert Kilickaya, Aykut Erdem, Nazli Ikizler-Cinbis, and Erkut Erdem. 2017. [Re-evaluating automatic metrics for image captioning](#). In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 1: Long Papers*, pages 199–209. Association for Computational Linguistics.
- Jin-Hwa Kim, Nikita Kitaev, Xinlei Chen, Marcus Rohrbach, Byoung-Tak Zhang, Yuandong Tian, Dhruv Batra, and Devi Parikh. 2019a. [Codraw: Collaborative drawing as a testbed for grounded goal-driven communication](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 6495–6513. Association for Computational Linguistics.
- Suwon Kim, HongYong Choi, JoongWon Hwang, JangYoung Song, SangRok Lee, TaeKang Woo, and AI Modulabs. 2019b. Vizwiz image captioning based on aoanet with scene graph. *ivc.ischool.utexas.edu*.

- Taehyeong Kim, Min-Oh Heo, Seonil Son, Kyoung-Wha Park, and Byoung-Tak Zhang. 2018. Glac net: Glocal attention cascading networks for multi-image cued story generation. *arXiv preprint arXiv:1805.10973*.
- Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. 2016. Character-aware neural language models. In *AAAI*, pages 2741–2749.
- Ben King and Steven Abney. 2013. Labeling the languages of words in mixed-language documents using weakly supervised methods. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1110–1119.
- Diederik P. Kingma and Jimmy Ba. 2014. [Adam: A method for stochastic optimization](#). *CoRR*, abs/1412.6980.
- Svetlana Kiritchenko and Saif Mohammad. 2018. [Examining gender and race bias in two hundred sentiment analysis systems](#). In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics, *SEM@NAACL-HLT 2018, New Orleans, Louisiana, USA, June 5-6, 2018*, pages 43–53. Association for Computational Linguistics.
- Jamie Ryan Kiros, William Chan, and Geoffrey E. Hinton. 2018. [Illustrative language understanding: Large-scale visual grounding with image search](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 922–933. Association for Computational Linguistics.
- Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M Rush. 2017. Opennmt: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72.
- Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86. Citeseer.
- Shimon Kogan, Tobias J Moskowicz, and Marina Niessner. 2019. Fake news: Evidence from financial markets. *Available at SSRN 3237763*.
- Eric Kolve, Roozbeh Mottaghi, Winson Han, Eli VanderBilt, Luca Weihs, Alvaro Herrasti, Daniel Gordon, Yuke Zhu, Abhinav Gupta, and Ali Farhadi. 2017. AI2-THOR: An Interactive 3D Environment for Visual AI. *arXiv*.
- Rik Koncel-Kedziorski, Dhanush Bekal, Yi Luan, Mirella Lapata, and Hannaneh Hajishirzi. 2019. [Text generation from knowledge graphs with graph transformers](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 2284–2293. Association for Computational Linguistics.
- Rik Koncel-Kedziorski, Hannaneh Hajishirzi, and Ali Farhadi. 2014. [Multi-resolution language grounding with weak supervision](#). In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 386–396. ACL.

- Ioannis Konstas and Mirella Lapata. 2012. Unsupervised concept-to-text generation with hypergraphs. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 752–761. Association for Computational Linguistics.
- Anaïs Koptient, Muriel Londres, and Natalia Grabar. 2021. [Fine-grained simplification of medical documents](#). In *Public Health and Informatics - Proceedings of MIE 2021, Medical Informatics Europe, Virtual Event, May 29-31, 2021*, volume 281 of *Studies in Health Technology and Informatics*, pages 308–312. IOS Press.
- Satwik Kottur, Seungwhan Moon, Alborz Geramifard, and Babak Damavandi. 2021a. [SIMMC 2.0: A task-oriented dialog dataset for immersive multimodal conversations](#). *CoRR*, abs/2104.08667.
- Satwik Kottur, Seungwhan Moon, Alborz Geramifard, and Babak Damavandi. 2021b. [SIMMC 2.0: A task-oriented dialog dataset for immersive multimodal conversations](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 4903–4912. Association for Computational Linguistics.
- Robert M Krauss and Susan R Fussell. 1990. Mutual knowledge and communicative effectiveness. *Intellectual teamwork: Social and technological foundations of cooperative work*, pages 111–146.
- Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael Bernstein, and Li Fei-Fei. 2017. Visual Genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 123(1):32–73.
- Reno Kriz, João Sedoc, Marianna Apidianaki, Carolina Zheng, Gaurav Kumar, Eleni Miltsakaki, and Chris Callison-Burch. 2019. Complexity-weighted loss and diverse reranking for sentence simplification. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3137–3147.
- Alexander Ku, Peter Anderson, Roma Patel, Eugene Ie, and Jason Baldridge. 2020. [Room-across-room: Multilingual vision-and-language navigation with dense spatiotemporal grounding](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 4392–4412. Association for Computational Linguistics.
- Iliia Kulikov, Alexander Miller, Kyunghyun Cho, and Jason Weston. 2019. Importance of search and evaluation strategies in neural dialogue modeling. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 76–87.
- Girish Kulkarni, Visruth Premraj, Vicente Ordonez, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C Berg, and Tamara L Berg. 2013. Babytalk: Understanding and generating simple image descriptions. volume 35, pages 2891–2903. IEEE.

- Anoop Kunchukuttan, Ratish Puduppully, and Pushpak Bhattacharyya. 2015. Brahmi-Net: A transliteration and script conversion system for languages of the Indian subcontinent. In *HLT-NAACL*, pages 81–85.
- Kiyoshi Kurihara, Atsushi Imai, Nobumasa Seiyama, Toshihiro Shimizu, Shoei Sato, Ichiro Yamada, Tadashi Kumano, Reiko Tako, Taro Miyazaki, Manon Ichiki, et al. 2019. Automatic generation of audio descriptions for sports programs. *SMPTE Motion Imaging Journal*, 128(1):41–47.
- Alina Kuznetsova, Mohamad Hassan Mohamad Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Mallocci, Alexander Kolesnikov, et al. 2020. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale.
- Polina Kuznetsova, Vicente Ordonez, Tamara L Berg, and Yejin Choi. 2014. Treetalk: Composition and compression of trees for image descriptions. *Transactions of the Association for Computational Linguistics*, 2:351–362.
- Himabindu Lakkaraju and Osbert Bastani. 2020. "how do I fool you?": Manipulating user trust via misleading black box explanations. In *AIES '20: AAAI/ACM Conference on AI, Ethics, and Society, New York, NY, USA, February 7-8, 2020*, pages 79–85. ACM.
- Alex M Lamb, Anirudh Goyal Alias Parth Goyal, Ying Zhang, Saizheng Zhang, Aaron C Courville, and Yoshua Bengio. 2016. Professor forcing: A new algorithm for training recurrent networks. In *Advances In Neural Information Processing Systems*, pages 4601–4609.
- Guillaume Lample, Sandeep Subramanian, Eric Smith, Ludovic Denoyer, Marc’Aurelio Ranzato, and Y-Lan Boureau. 2019. [Multiple-attribute text rewriting](#). In *International Conference on Learning Representations*.
- Weiyu Lan, Xirong Li, and Jianfeng Dong. 2017. Fluency-guided cross-lingual image captioning. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1549–1557.
- Veronica Latcinnik and Jonathan Berant. 2020. [Explaining question answering models through text generation](#). *CoRR*, abs/2004.05569.
- Javier Latorre, Koji Iwano, and Sadaoki Furui. 2006. New approach to the polyglot speech generation by means of an HMM-based speaker adaptable synthesizer. *Speech Communication*, 48(10):1227–1242.
- Florian Laws, Lukas Michelbacher, Beate Dorow, Christian Scheible, Ulrich Heid, and Hinrich Schütze. 2010. [A linguistically grounded graph model for bilingual lexicon extraction](#). In *COLING 2010, 23rd International Conference on Computational Linguistics, Posters Volume, 23-27 August 2010, Beijing, China*, pages 614–622. Chinese Information Processing Society of China.
- Chris van der Lee, Albert Gatt, Emiel van Miltenburg, Sander Wubben, and Emiel Kraemer. 2019. [Best practices for the human evaluation of automatically generated text](#). In *Proceedings of the 12th International Conference on Natural Language Generation, INLG 2019, Tokyo, Japan, October 29 - November 1, 2019*, pages 355–368. Association for Computational Linguistics.

- Grandee Lee, Xianghu Yue, and Haizhou Li. 2019. Linguistically motivated parallel data augmentation for code-switch language modeling. *Proc. Interspeech 2019*, pages 3730–3734.
- Jason Lee, Elman Mansimov, and Kyunghyun Cho. 2018. [Deterministic non-autoregressive neural sequence modeling by iterative refinement](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 1173–1182. Association for Computational Linguistics.
- Wendy G Lehnert. 1982. Plot units: A narrative summarization strategy. *Strategies for natural language processing*, pages 375–414.
- Wendy G Lehnert and Elaine W Vine. 1987. The role of affect in narrative structure. *Cognition and emotion*, 1(3):299–322.
- Jie Lei, Licheng Yu, Tamara L. Berg, and Mohit Bansal. 2020a. [TVQA+: spatio-temporal grounding for video question answering](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 8211–8225. Association for Computational Linguistics.
- Jie Lei, Licheng Yu, Tamara L. Berg, and Mohit Bansal. 2020b. [What is more likely to happen next? video-and-language future event prediction](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 8769–8784. Association for Computational Linguistics.
- David Lewis. 2008. *Convention: A philosophical study*. John Wiley & Sons.
- Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.
- Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. [Retrieval-augmented generation for knowledge-intensive NLP tasks](#). *CoRR*, abs/2005.11401.
- Gen Li, Nan Duan, Yuejian Fang, Daxin Jiang, and Ming Zhou. 2020a. Unicoder-VL: A universal encoder for vision and language by cross-modal pre-training. In *AAAI*.
- Guang Li, Linchao Zhu, Ping Liu, and Yi Yang. 2019a. [Entangled transformer for image captioning](#). In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 8927–8936. IEEE.
- Jiangyun Li, Peng Yao, Longteng Guo, and Weicun Zhang. 2019b. Boosted transformer for image captioning. *Applied Sciences (2076-3417)*, 9(16).
- Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016a. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119.

- Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. 2016b. A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 994–1003.
- Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. 2017. [Adversarial learning for neural dialogue generation](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 2157–2169. Association for Computational Linguistics.
- Juncen Li, Robin Jia, He He, and Percy Liang. 2018a. Delete, retrieve, generate: a simple approach to sentiment and style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1865–1874.
- Nannan Li and Zhenzhong Chen. 2018. [Image captioning with visual-semantic LSTM](#). In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 793–799. ijcai.org.
- Qing Li, Qingyi Tao, Shafiq R. Joty, Jianfei Cai, and Jiebo Luo. 2018b. [VQA-E: explaining, elaborating, and enhancing your answers for visual questions](#). In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VII*, volume 11211 of *Lecture Notes in Computer Science*, pages 570–586. Springer.
- Siming Li, Girish Kulkarni, Tamara Berg, Alexander Berg, and Yejin Choi. 2011. Composing simple image descriptions using web-scale n-grams. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, pages 220–228.
- Wei Li, Can Gao, Guocheng Niu, Xinyan Xiao, Hao Liu, Jiachen Liu, Hua Wu, and Haifeng Wang. 2021. [UNIMO: towards unified-modal understanding and generation via cross-modal contrastive learning](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 2592–2607. Association for Computational Linguistics.
- Xiang Lisa Li and Percy Liang. 2021. [Prefix-tuning: Optimizing continuous prompts for generation](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 4582–4597. Association for Computational Linguistics.
- Xiaolong Li and Kristy Boyer. 2015. [Semantic grounding in dialogue for complex problem solving](#). In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 841–850. The Association for Computational Linguistics.
- Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. 2020b. [Oscar: Object-semantics aligned pre-training for vision-language tasks](#). In *Computer Vision - ECCV 2020 -*

- 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XXX, volume 12375 of *Lecture Notes in Computer Science*, pages 121–137. Springer.
- Yang Li and Peter Clark. 2015. [Answering elementary science questions by constructing coherent scenes using background knowledge](#). In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 2007–2012. The Association for Computational Linguistics.
- Ying Li and Pascale Fung. 2012. Code-switch language model with inversion constraints for mixed language speech recognition. In *Proceedings of COLING 2012*, pages 1671–1680.
- Ying Li and Pascale Fung. 2013. Improved mixed language speech recognition using asymmetric acoustic model and language model with code-switch inversion constraints. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 7368–7372. IEEE.
- Ying Li and Pascale Fung. 2014. Code switch language modeling with functional head constraint. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 4913–4917. IEEE.
- Yutong Li, Nicholas Gekakis, Qiuzhe Wu, Boyue Li, Khyathi Chandu, and Eric Nyberg. 2018c. Extraction meets abstraction: Ideal answer generation for biomedical questions. In *Proceedings of the 6th BioASQ Workshop A challenge on large-scale biomedical semantic indexing and question answering*, pages 57–65.
- Zekang Li, Cheng Niu, Fandong Meng, Yang Feng, Qian Li, and Jie Zhou. 2019c. [Incremental transformer with deliberation decoder for document grounded conversations](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 12–21. Association for Computational Linguistics.
- Hui Liang, Yao Qian, and Frank K Soong. 2007. An HMM-based Bilingual (Mandarin-English) TTS. *Proceedings of SSW6*.
- Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*.
- Chin-Yew Lin and Eduard Hovy. 2002. Manual and automatic evaluation of summaries. In *Proceedings of the ACL-02 Workshop on Automatic Summarization*, pages 45–51.
- Chin-Yew Lin and Eduard Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 71–78. Association for Computational Linguistics.
- Hui Lin and Vincent Ng. 2019. Abstractive summarization: A survey of the state of the art. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9815–9822.
- Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. [Microsoft COCO: common objects in context](#). In *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September*

- 6-12, 2014, *Proceedings, Part V*, volume 8693 of *Lecture Notes in Computer Science*, pages 740–755. Springer.
- Sander Van der Linden, Costas Panagopoulos, and Jon Roozenbeek. 2020. You are fake news: political bias in perceptions of fake news. *Media, Culture & Society*, 42(3):460–470.
- Zachary C. Lipton. 2018. [The mythos of model interpretability](#). *Commun. ACM*, 61(10):36–43.
- Changsong Liu, Lanbo She, Rui Fang, and Joyce Y. Chai. 2014. [Probabilistic labeling for efficient referential grounding based on collaborative discourse](#). In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 2: Short Papers*, pages 13–18. The Association for Computer Linguistics.
- Changsong Liu, Shaohua Yang, Sari Saba-Sadiya, Nishant Shukla, Yunzhong He, Song-Chun Zhu, and Joyce Yue Chai. 2016a. [Jointly learning grounded task structures from language instruction and visual demonstration](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1482–1492. The Association for Computational Linguistics.
- Chia-Wei Liu, Ryan Lowe, Iulian Vlad Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. 2016b. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2122–2132.
- Fenglin Liu, Xuancheng Ren, Yuanxin Liu, Houfeng Wang, and Xu Sun. 2018a. [simnet: Step-wise image-topic merging network for generating detailed and comprehensive image captions](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 137–149.
- Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. [Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing](#). *CoRR*, abs/2107.13586.
- Peter J Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. 2018b. Generating wikipedia by summarizing long sequences. *arXiv preprint arXiv:1801.10198*.
- Siqi Liu, Zhenhai Zhu, Ning Ye, Sergio Guadarrama, and Kevin Murphy. 2017a. [Improved image captioning via policy gradient optimization of spider](#). In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 873–881. IEEE Computer Society.
- Xiaoxiao Liu, Qingyang Xu, and Ning Wang. 2019. A survey on deep neural network-based image captioning. *The Visual Computer*, 35(3):445–470.
- Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. [Multilingual denoising pre-training for neural machine translation](#). *Trans. Assoc. Comput. Linguistics*, 8:726–742.

- Yu Liu, Jianlong Fu, Tao Mei, and Chang Wen Chen. 2017b. Let your photos talk: Generating narrative paragraph for photo stream via bidirectional attention recurrent neural networks. In *Thirty-First AAAI Conference on Artificial Intelligence*.
- Chi-kiu Lo. 2017. [MEANT 2.0: Accurate semantic MT evaluation for any output language](#). In *Proceedings of the Second Conference on Machine Translation, WMT 2017, Copenhagen, Denmark, September 7-8, 2017*, pages 589–597. Association for Computational Linguistics.
- Chi-kiu Lo, Michel Simard, Darlene A. Stewart, Samuel Larkin, Cyril Goutte, and Patrick Littell. 2018. [Accurate semantic textual similarity for cleaning noisy parallel corpora using semantic machine translation evaluation metric: The NRC supervised submissions to the parallel corpus filtering task](#). In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers, WMT 2018, Belgium, Brussels, October 31 - November 1, 2018*, pages 908–916. Association for Computational Linguistics.
- Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. 2020. [S2ORC: The semantic scholar open research corpus](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983, Online. Association for Computational Linguistics.
- Lajanugen Logeswaran, Honglak Lee, and Samy Bengio. 2018. Content preserving text generation with attribute controls. In *Advances in Neural Information Processing Systems*, pages 5103–5113.
- Ryan Lowe, Michael Noseworthy, Iulian Vlad Serban, Nicolas Angelard-Gontier, Yoshua Bengio, and Joelle Pineau. 2017. [Towards an automatic turing test: Learning to evaluate dialogue responses](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1116–1126. Association for Computational Linguistics.
- Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019a. ViLBERT: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*.
- Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019b. ViLBERT: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*.
- Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. 2017. [Knowing when to look: Adaptive attention via a visual sentinel for image captioning](#). In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 3242–3250. IEEE Computer Society.
- Sidi Lu, Yaoming Zhu, Weinan Zhang, Jun Wang, and Yong Yu. 2018a. Neural text generation: past, present and beyond. *arXiv preprint arXiv:1803.07133*.
- Sidi Lu, Yaoming Zhu, Weinan Zhang, Jun Wang, and Yong Yu. 2018b. [Neural text generation: Past, present and beyond](#). *CoRR*, abs/1803.07133.
- Ximing Lu, Peter West, Rowan Zellers, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. [Neurologic decoding: \(un\)supervised neural text generation with predicate logic constraints](#). *CoRR*, abs/2010.12884.

- Yi Luan, Chris Brockett, Bill Dolan, Jianfeng Gao, and Michel Galley. 2017. [Multi-task learning for speaker-role adaptation in neural conversation models](#). In *Proceedings of the Eighth International Joint Conference on Natural Language Processing, IJCNLP 2017, Taipei, Taiwan, November 27 - December 1, 2017 - Volume 1: Long Papers*, pages 605–614. Asian Federation of Natural Language Processing.
- Stephanie Lukin, Reginald Hobbs, and Clare Voss. 2018. A pipeline for creative visual storytelling. In *Proceedings of the First Workshop on Storytelling*, pages 20–32.
- Ruotian Luo and Greg Shakhnarovich. 2020. [Controlling length in image captioning](#). *CoRR*, abs/2005.14386.
- Minh-Thang Luong, Michael C. Frank, and Mark Johnson. 2013. [Parsing entire discourses as very long strings: Capturing topic continuity in grounded language learning](#). *Trans. Assoc. Comput. Linguistics*, 1:315–326.
- Dau-Cheng Lyu, Tien-Ping Tan, Eng Siong Chng, and Haizhou Li. 2010. Seame: a mandarin-english code-switching speech corpus in south-east asia. In *Eleventh Annual Conference of the International Speech Communication Association*.
- Qingsong Ma, Yvette Graham, Shugen Wang, and Qun Liu. 2017. [Blend: a novel combined MT metric based on direct assessment - CASICT-DCU submission to WMT17 metrics task](#). In *Proceedings of the Second Conference on Machine Translation, WMT 2017, Copenhagen, Denmark, September 7-8, 2017*, pages 598–603. Association for Computational Linguistics.
- François Mairesse and Marilyn A Walker. 2010. Towards personality-based user adaptation: psychologically informed stylistic language generation. *User Modeling and User-Adapted Interaction*, 20(3):227–278.
- Bodhisattwa Prasad Majumder, Shuyang Li, Jianmo Ni, and Julian J. McAuley. 2020. [Interview: Large-scale modeling of media dialog with discourse patterns and knowledge grounding](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 8129–8141. Association for Computational Linguistics.
- Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60.
- Thomas Manzini, Yao Chong Lim, Alan W. Black, and Yulia Tsvetkov. 2019. [Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 615–621. Association for Computational Linguistics.
- Mitchell Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. [The penn treebank: Annotating predicate argument structure](#). In *Proceedings of the Workshop on Human Language Technology, HLT '94*, pages 114–119, Stroudsburg, PA, USA. Association for Computational Linguistics.

- Javier Marin, Aritro Biswas, Ferda Ofli, Nicholas Hynes, Amaia Salvador, Yusuf Aytar, Ingmar Weber, and Antonio Torralba. 2018. Recipe1m: A dataset for learning cross-modal embeddings for cooking recipes and food images. *arXiv preprint arXiv:1810.06553*.
- Lara J Martin, Prithviraj Ammanabrolu, Xinyu Wang, William Hancock, Shruti Singh, Brent Harrison, and Mark O Riedl. 2017. Event representations for automated story generation with deep neural nets. *arXiv preprint arXiv:1706.01331*.
- Lara J Martin, Prithviraj Ammanabrolu, Xinyu Wang, William Hancock, Shruti Singh, Brent Harrison, and Mark O Riedl. 2018. Event representations for automated story generation with deep neural nets. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Luca Massarelli, Fabio Petroni, Aleksandra Piktus, Myle Ott, Tim Rocktäschel, Vassilis Plachouras, Fabrizio Silvestri, and Sebastian Riedel. 2019. How decoding strategies affect the verifiability of generated text. *arXiv preprint arXiv:1911.03587*.
- Alexandre Blondin Massé, Guillaume Chicoisne, Yassine Gargouri, Stevan Harnad, Odile Marcotte, and Olivier Picard. 2008. How is meaning grounded in dictionary definitions? In *Coling 2008: Proceedings of the 3rd Textgraphs workshop on Graph-based Algorithms for Natural Language Processing*, pages 17–24.
- Alexander Mathews, Lexing Xie, and Xuming He. 2018. Semstyle: Learning to generate stylised image captions using unaligned text. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8591–8600.
- Puneet Mathur, Rajiv Shah, Ramit Sawhney, and Debanjan Mahata. 2018. Detecting offensive tweets in hindi-english code-switched language. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*, pages 18–26.
- Stephen Mayhew, Klinton Bicknell, Chris Brust, Bill McDowell, Will Monroe, and Burr Settles. 2020. [Simultaneous translation and paraphrase for language education](#). In *Proceedings of the Fourth Workshop on Neural Generation and Translation, NGT@ACL 2020, Online, July 5-10, 2020*, pages 232–243. Association for Computational Linguistics.
- Brian McMahan and Matthew Stone. 2015. [A bayesian model of grounded color semantics](#). *Trans. Assoc. Comput. Linguistics*, 3:103–115.
- Stephen Merity, Nitish Shirish Keskar, and Richard Socher. 2017. Regularizing and optimizing lstm language models. *arXiv preprint arXiv:1708.02182*.
- Stephen Merity, Nitish Shirish Keskar, and Richard Socher. 2018. An analysis of neural language modeling at multiple scales. *arXiv preprint arXiv:1803.08240*.
- Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2016. [Pointer sentinel mixture models](#). *CoRR*, abs/1609.07843.
- Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černocký, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Eleventh Annual Conference of the International Speech Communication Association*.

- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.
- George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.
- James Milroy et al. 1995. *One speaker, two languages: Cross-disciplinary perspectives on code-switching*. Cambridge University Press.
- Dipendra Kumar Misra, Andrew Bennett, Valts Blukis, Eyvind Niklasson, Max Shatkhin, and Yoav Artzi. 2018. [Mapping instructions to actions in 3d environments with visual goal prediction](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2667–2678. Association for Computational Linguistics.
- Dan I Moldovan, Sanda M Harabagiu, Roxana Girju, Paul Morarescu, V Finley Lacatusu, Adrian Novischi, Adriana Badulescu, and Orest Bolohan. 2002. Lcc tools for question answering. In *TREC*.
- Giovanni Molina, Fahad AlGhamdi, Mahmoud Ghoneim, Abdelati Hawwari, Nicolas Rey-Villamizar, Mona Diab, and Tamar Solorio. 2016. Overview for the second shared task on language identification in code-switched data. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pages 40–49.
- Will Monroe, Robert X. D. Hawkins, Noah D. Goodman, and Christopher Potts. 2017. [Colors in context: A pragmatic neural model for grounded language understanding](#). *Trans. Assoc. Comput. Linguistics*, 5:325–338.
- Joao Luis Zeni Montenegro, Cristiano André da Costa, and Rodrigo da Rosa Righi. 2019. Survey of conversational agents in health. *Expert Systems with Applications*.
- Eva M Moreno, Kara D Federmeier, and Marta Kutas. 2002. Switching languages, switching palabras (words): An electrophysiological study of code switching. *Brain and language*, 80(2):188–207.
- Shinsuke Mori, Hirokuni Maeta, Yoko Yamakata, and Tetsuro Sasada. 2014. Flow graph corpus from recipe texts. In *LREC*, pages 2370–2377.
- Nasrin Mostafazadeh, Chris Brockett, Bill Dolan, Michel Galley, Jianfeng Gao, Georgios P Spithourakis, and Lucy Vanderwende. 2017. Image-grounded conversations: Multimodal context for natural question and response generation. *arXiv preprint arXiv:1701.08251*.
- Nasrin Mostafazadeh, Alyson Grealish, Nathanael Chambers, James Allen, and Lucy Vanderwende. 2016. [CaTeRS: Causal and temporal relation scheme for semantic annotation of event structures](#). In *Proceedings of the Fourth Workshop on Events*, pages 51–61, San Diego, California. Association for Computational Linguistics.
- Lili Mou, Yiping Song, Rui Yan, Ge Li, Lu Zhang, and Zhi Jin. 2016. [Sequence to backward and forward sequences: A content-introducing approach to generative short-text conversation](#). In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 3349–3358. ACL.

- Janet H. Murray. 2015. [Tell me a story: Toward more expressive and coherent computational narratives \(invited talk\)](#). In *6th Workshop on Computational Models of Narrative, CMN 2015, May 26-28, 2015, Atlanta, GA, USA*, volume 45 of *OASICS*, pages 1–1. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.
- Pieter Muysken. 2000. *Bilingual speech: A typology of code-mixing*, volume 11. Cambridge University Press.
- Carol Myers-Scotton. 1997. *Duelling languages: Grammatical structure in codeswitching*. Oxford University Press.
- Yukiko I. Nakano, Gabe Reinstein, Tom Stocky, and Justine Cassell. 2003. [Towards a model of face-to-face grounding](#). In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, 7-12 July 2003, Sapporo Convention Center, Sapporo, Japan*, pages 553–561. ACL.
- Ramesh Nallapati, Bowen Zhou, and Mingbo Ma. 2016. Classify or select: Neural architectures for extractive document summarization. *arXiv preprint arXiv:1611.04244*.
- Sushobhan Nayak and Amitabha Mukerjee. 2012. [Grounded language acquisition: A minimal commitment approach](#). In *COLING 2012, 24th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, 8-15 December 2012, Mumbai, India*, pages 2059–2076. Indian Institute of Technology Bombay.
- Rani Nelken and Elif Yamangil. 2008. Mining wikipedia’s article revision history for training computational linguistics algorithms. In *Proceedings of the AAAI Workshop on Wikipedia and Artificial Intelligence: An Evolving Synergy*, pages 31–36.
- Ani Nenkova and Kathleen McKeown. 2012. A survey of text summarization techniques. In *Mining text data*, pages 43–76. Springer.
- Ani Nenkova and Rebecca J. Passonneau. 2004. [Evaluating content selection in summarization: The pyramid method](#). In *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, HLT-NAACL 2004, Boston, Massachusetts, USA, May 2-7, 2004*, pages 145–152. The Association for Computational Linguistics.
- Ani Nenkova, Rebecca J. Passonneau, and Kathleen R. McKeown. 2007. [The pyramid method: Incorporating human content selection variation in summarization evaluation](#). *ACM Trans. Speech Lang. Process.*, 4(2):4.
- Khanh Nguyen and Hal Daumé III. 2019. Help, anna! visual navigation with natural multimodal assistance via retrospective curiosity-encouraging imitation learning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 684–695.
- Minheng Ni, Haoyang Huang, Lin Su, Edward Cui, Taroon Bharti, Lijuan Wang, Dongdong Zhang, and Nan Duan. 2021. [M3P: learning universal representations via multitask multilingual multimodal pre-training](#). In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 3977–3986. Computer Vision Foundation / IEEE.

- Joel Nothman, Matthew Honnibal, Ben Hachey, and James R. Curran. 2012. [Event linking: Grounding event reference in a news archive](#). In *The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, July 8-14, 2012, Jeju Island, Korea - Volume 2: Short Papers*, pages 228–232. The Association for Computer Linguistics.
- Tim Oates. 2003. Grounding word meanings in sensor data: Dealing with referential uncertainty. In *Proceedings of the HLT-NAACL 2003 workshop on Learning word meaning from non-linguistic data*, pages 62–69.
- Aäron van den Oord, Yazhe Li, Igor Babuschkin, Karen Simonyan, Oriol Vinyals, Koray Kavukcuoglu, George van den Driessche, Edward Lockhart, Luis C. Cobo, Florian Stimberg, Norman Casagrande, Dominik Grewe, Seb Noury, Sander Dieleman, Erich Elsen, Nal Kalchbrenner, Heiga Zen, Alex Graves, Helen King, Tom Walters, Dan Belov, and Demis Hassabis. 2018. [Parallel wavenet: Fast high-fidelity speech synthesis](#). In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 3915–3923. PMLR.
- Susan K Opt. 1988. Continuity and change in storytelling about artificial intelligence: Extending the narrative paradigm. *Communication Quarterly*, 36(4):298–310.
- Jessica Ouyang and Kathleen R. McKeown. 2015. [Modeling reportable events as turning points in narrative](#). In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 2149–2158. The Association for Computational Linguistics.
- Aishwarya Padmakumar, Jesse Thomason, Ayush Shrivastava, Patrick Lange, Anjali Narayan-Chen, Spandana Gella, Robinson Piramuthu, Gökhan Tür, and Dilek Hakkani-Tür. 2021. [Teach: Task-driven embodied agents that chat](#). *CoRR*, abs/2110.00534.
- Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2018. Unsupervised learning of sentence embeddings using compositional n-gram features. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 528–540.
- Brian E. Pangburn, S. Sitharama Iyengar, Robert C. Mathews, and Jonathan P. Ayo. 2003. [EBLA: A perceptually grounded model of language acquisition](#). In *Proceedings of the HLT-NAACL 2003 Workshop on Learning Word Meaning from Non-Linguistic Data*, pages 46–53.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002a. [Bleu: a method for automatic evaluation of machine translation](#). In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318. ACL.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002b. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

- Nikolaos Pappas, Phoebe Mulcaire, and Noah A. Smith. 2020. [Grounded compositional outputs for adaptive language modeling](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 1252–1267. Association for Computational Linguistics.
- Ankur P. Parikh, Hoifung Poon, and Kristina Toutanova. 2015. [Grounded semantic parsing for complex knowledge extraction](#). In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 756–766. The Association for Computational Linguistics.
- Cesc C Park and Gunhee Kim. 2015. Expressing an image stream with a sequence of natural sentences. In *Advances in neural information processing systems*, pages 73–81.
- Alok Parlikar. 2012. TestVox: web-based framework for subjective evaluation of speech synthesis. *Opensource Software*.
- Alok Parlikar, Sunayana Sitaram, Andrew Wilkinson, and Alan W Black. 2016. The festvox indic frontend for grapheme to phoneme conversion. In *WILDRE: Workshop on Indian Language Data-Resources and Evaluation*.
- Jasabanta Patro, Bidisha Samanta, Saurabh Singh, Abhipsa Basu, Prithwish Mukherjee, Monojit Choudhury, and Animesh Mukherjee. 2017. All that is english may be hindi: Enhancing language identification through automatic ranking of the likeliness of word borrowing in social media. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2264–2274.
- Romain Paulus, Caiming Xiong, and Richard Socher. 2018. [A deep reinforced model for abstractive summarization](#). In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.
- Obioma Pelka, Sven Koitka, Johannes Rückert, Felix Nensa, and Christoph M. Friedrich. 2018. [Radiology objects in context \(ROCO\): A multimodal image dataset](#). In *Intravascular Imaging and Computer Assisted Stenting - and - Large-Scale Annotation of Biomedical Data and Expert Label Synthesis - 7th Joint International Workshop, CVII-STENT 2018 and Third International Workshop, LABELS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 16, 2018, Proceedings*, volume 11043 of *Lecture Notes in Computer Science*, pages 180–189. Springer.
- Nanyun Peng, Marjan Ghazvininejad, Jonathan May, and Kevin Knight. 2018. Towards controllable story generation. In *Proceedings of the First Workshop on Storytelling*, pages 43–49.
- Rivindu Perera and Parma Nand. 2017. [Recent advances in natural language generation: A survey and classification of the empirical literature](#). *Comput. Informatics*, 36(1):1–32.
- Laura Perez-Beltrachini and Mirella Lapata. 2018. Bootstrapping generators from noisy data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1516–1527.

- Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237.
- Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick S. H. Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander H. Miller. 2019. [Language models as knowledge bases?](#) In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2463–2473. Association for Computational Linguistics.
- Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015a. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pages 2641–2649.
- Bryan A. Plummer, Liwei Wang, Chris M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015b. [Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models.](#) In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 2641–2649. IEEE Computer Society.
- Hoifung Poon. 2013. [Grounded unsupervised semantic parsing.](#) In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers*, pages 933–943. The Association for Computer Linguistics.
- Shana Poplack. 1980. Sometimes i’ll start a sentence in spanish y termino en espanol: toward a typology of code-switching1. *Linguistics*, 18(7-8):581–618.
- Vinodkumar Prabhakaran, Ben Hutchinson, and Margaret Mitchell. 2019. [Perturbation sensitivity analysis to detect unintended model biases.](#) In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 5739–5744. Association for Computational Linguistics.
- Shrimai Prabhumoye, Alan W Black, and Ruslan Salakhutdinov. 2020. Exploring controllable text generation techniques. *arXiv preprint arXiv:2005.01822*.
- Shrimai Prabhumoye, Chris Quirk, and Michel Galley. 2019. [Towards content transfer through grounded text generation.](#) In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 2622–2632. Association for Computational Linguistics.
- Shrimai Prabhumoye, Yulia Tsvetkov, Ruslan Salakhutdinov, and Alan W Black. 2018. Style transfer through back-translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 866–876.

- Adithya Pratapa, Gayatri Bhat, Monojit Choudhury, Sunayana Sitaram, Sandipan Dandapat, and Kalika Bali. 2018. Language modeling for code-mixing: The role of linguistic theory based synthetic data. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1543–1553.
- Marcelo O. R. Prates, Pedro H. C. Avelar, and Luís C. Lamb. 2020. [Assessing gender bias in machine translation: a case study with google translate](#). *Neural Comput. Appl.*, 32(10):6363–6381.
- Ratish Puduppully, Li Dong, and Mirella Lapata. 2018. Data-to-text generation with content selection and planning. *arXiv preprint arXiv:1809.00582*.
- Ratish Puduppully, Li Dong, and Mirella Lapata. 2019. Data-to-text generation with content selection and planning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6908–6915.
- Xavier Puig, Kevin Ra, Marko Boben, Jiaman Li, Tingwu Wang, Sanja Fidler, and Antonio Torralba. 2018. [Virtualhome: Simulating household activities via programs](#). In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 8494–8502. Computer Vision Foundation / IEEE Computer Society.
- Guanghui Qin, Jin-Ge Yao, Xuening Wang, Jinpeng Wang, and Chin-Yew Lin. 2018. [Learning latent semantic annotations for grounding natural language to structured data](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 3761–3771. Association for Computational Linguistics.
- Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training.
- Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. [Exploring the limits of transfer learning with a unified text-to-text transformer](#). *CoRR*, abs/1910.10683.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. [Exploring the limits of transfer learning with a unified text-to-text transformer](#). *J. Mach. Learn. Res.*, 21:140:1–140:67.
- Kiran Ramesh, Surya Ravishankaran, Abhishek Joshi, and K Chandrasekaran. 2017. A survey of design techniques for conversational agents. In *International Conference on Information, Communication and Computing Technology*, pages 336–350. Springer.
- Marc’Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2016. [Sequence level training with recurrent neural networks](#). In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*.

- Sudha Rao and Joel Tetreault. 2018. Dear sir or madam, may i introduce the gyafc dataset: Corpus, benchmarks and metrics for formality style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 129–140.
- Michaela Regneri, Marcus Rohrbach, Dominikus Wetzels, Stefan Thater, Bernt Schiele, and Manfred Pinkal. 2013. [Grounding action descriptions in videos](#). *Trans. Assoc. Comput. Linguistics*, 1:25–36.
- Ehud Reiter. 2018. [A structured review of the validity of BLEU](#). *Comput. Linguistics*, 44(3).
- Ehud Reiter and Anja Belz. 2009. An investigation into the validity of some metrics for automatically evaluating natural language generation systems. *Computational Linguistics*, 35(4):529–558.
- Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. 2015. [Faster R-CNN: towards real-time object detection with region proposal networks](#). In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 91–99.
- Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. [Self-critical sequence training for image captioning](#). In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 1179–1195. IEEE Computer Society.
- Mark O. Riedl and Brent Harrison. 2016. [Using stories to teach human values to artificial agents](#). In *AI, Ethics, and Society, Papers from the 2016 AAAI Workshop, Phoenix, Arizona, USA, February 13, 2016*, volume WS-16-02 of *AAAI Workshops*. AAAI Press.
- Elena Rishes, Stephanie M Lukin, David K Elson, and Marilyn A Walker. 2013. Generating different story tellings from semantic representations of narrative. In *International Conference on Interactive Digital Storytelling*, pages 192–204. Springer.
- Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. 2018. [Object hallucination in image captioning](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 4035–4045. Association for Computational Linguistics.
- Eleanor Rosch. 1975. Cognitive representations of semantic categories. *Journal of experimental psychology: General*, 104(3):192.
- Eleanor Rosch, Carolyn B Mervis, Wayne D Gray, David M Johnson, and Penny Boyes-Braem. 1976. Basic objects in natural categories. *Cognitive psychology*, 8(3):382–439.
- Candace Ross, Andrei Barbu, Yevgeni Berzak, Battushig Myanganbayar, and Boris Katz. 2018. [Grounding language acquisition by training semantic parsers using captioned videos](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2647–2656. Association for Computational Linguistics.

- Aurko Roy, Ashish Vaswani, Arvind Neelakantan, and Niki Parmar. 2018. [Theory and experiments on vector quantized autoencoders](#). *CoRR*, abs/1805.11063.
- Deb Roy, Kai-Yuh Hsiao, and Nikolaos Mavridis. 2003. Conversational robots: building blocks for grounding word meaning. In *Proceedings of the HLT-NAACL 2003 workshop on Learning word meaning from non-linguistic data*, pages 70–77.
- Subhro Roy, Michael Noseworthy, Rohan Paul, Daehyung Park, and Nicholas Roy. 2019. [Leveraging past references for robust language grounding](#). In *Proceedings of the 23rd Conference on Computational Natural Language Learning, CoNLL 2019, Hong Kong, China, November 3-4, 2019*, pages 430–440. Association for Computational Linguistics.
- Subhro Roy, Shyam Upadhyay, and Dan Roth. 2016. [Equation parsing : Mapping sentences to grounded equations](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 1088–1097. The Association for Computational Linguistics.
- Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*.
- Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. 2018. [Gender bias in coreference resolution](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 8–14. Association for Computational Linguistics.
- Koustav Rudra, Shruti Rijhwani, Rafiya Begum, Kalika Bali, Monojit Choudhury, and Niloy Ganguly. 2016. Understanding language preference for expression of opinion and sentiment: What do hindi-english speakers do on twitter? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1131–1141.
- Vasile Rus, Zhiqiang Cai, and Arthur C. Graesser. 2007. [Experiments on generating questions about facts](#). In *Computational Linguistics and Intelligent Text Processing, 8th International Conference, CICLing 2007, Mexico City, Mexico, February 18-24, 2007*, volume 4394 of *Lecture Notes in Computer Science*, pages 444–455. Springer.
- Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015a. ImageNet large scale visual recognition challenge. *IJCV*, 115(3):211–252.
- Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015b. [ImageNet Large Scale Visual Recognition Challenge](#). *International Journal of Computer Vision (IJCV)*, 115(3):211–252.
- Alicia Sagae, Jerry R Hobbs, Suzanne Wertheim, Michael H Agar, Emily Ho, and W Lewis Johnson. 2012. Efficient cultural models of verbal behavior for communicative agents. In *International Conference on Intelligent Virtual Agents*, pages 523–525. Springer.

- Amaia Salvador, Michal Drozdal, Xavier Giró-i-Nieto, and Adriana Romero. 2019. [Inverse cooking: Recipe generation from food images](#). In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 10453–10462. Computer Vision Foundation / IEEE.
- Amaia Salvador, Michal Drozdal, Xavier Giro-i Nieto, and Adriana Romero. 2018. Inverse cooking: Recipe generation from food images. *arXiv preprint arXiv:1812.06164*.
- Bidisha Samanta, Sharmila Reddy, Hussain Jagirdar, Niloy Ganguly, and Soumen Chakrabarti. 2019. A deep generative model for code-switched text. *arXiv preprint arXiv:1906.08972*.
- Victor Sanh, Thomas Wolf, and Sebastian Ruder. 2018. A hierarchical multi-task approach for learning embeddings from semantic tasks. *arXiv preprint arXiv:1811.06031*.
- David Sankoff and Shana Poplack. 1981. A formal grammar for code-switching. *Research on Language & Social Interaction*, 14(1):3–45.
- Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A. Smith, and Yejin Choi. 2019. [ATOMIC: an atlas of machine commonsense for if-then reasoning](#). In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 3027–3035. AAAI Press.
- Maarten Sap, Vered Shwartz, Antoine Bosselut, Yejin Choi, and Dan Roth. 2020. [Commonsense reasoning for natural language processing](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts, ACL 2020, Online, July 5, 2020*, pages 27–33. Association for Computational Linguistics.
- Roger C Schank. 1990. *Tell me a story: A new look at real and artificial memory*. Charles Scribner’s Sons.
- Natalie Schilling. 2013. Investigating stylistic variation. *The handbook of language variation and change*, pages 325–349.
- Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. 2008. *Introduction to information retrieval*, volume 39. Cambridge University Press.
- Peri L Schuyler, William T Hole, Mark S Tuttle, and David D Sherertz. 1993. The umls metathesaurus: representing different views of biomedical concepts. *Bulletin of the Medical Library Association*, 81(2):217.
- Thomas Scialom, Paul-Alexis Dray, Patrick Gallinari, Sylvain Lamprier, Benjamin Piwowarski, Jacopo Staiano, and Alex Wang. 2021. [Questeval: Summarization asks for fact-based evaluation](#). *arXiv preprint arXiv:2103.12693*.
- Thomas Scialom, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. 2019. [Answers unite! unsupervised metrics for reinforced summarization models](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3244–3254. Association for Computational Linguistics.

- Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. [Get to the point: Summarization with pointer-generator networks](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1073–1083. Association for Computational Linguistics.
- Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. 2020. [BLEURT: learning robust metrics for text generation](#). *CoRR*, abs/2004.04696.
- Paul Hongsuck Seo, Piyush Sharma, Tomer Levinboim, Bohyung Han, and Radu Soricut. 2020. [Reinforcing an image caption generator using off-line human feedback](#). In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 2693–2700. AAAI Press.
- Lei Sha, Lili Mou, Tianyu Liu, Pascal Poupart, Sujian Li, Baobao Chang, and Zhifang Sui. 2018. Order-planning neural text generation from structured data. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Himanshu Sharma, Manmohan Agrahari, Sujeet Kumar Singh, Mohd Firoj, and Ravi Kumar Mishra. 2020. Image captioning: A comprehensive survey. In *2020 International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC)*, pages 325–328. IEEE.
- Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. [Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2556–2565. Association for Computational Linguistics.
- Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2017a. Style transfer from non-parallel text by cross-alignment. In *Advances in neural information processing systems*, pages 6830–6841.
- Xiaoyu Shen, Hui Su, Yanran Li, Wenjie Li, Shuzi Niu, Yang Zhao, Akiko Aizawa, and Guoping Long. 2017b. [A conditional variational framework for dialog generation](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pages 504–509. Association for Computational Linguistics.
- Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. 2019. [The woman worked as a babysitter: On biases in language generation](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3405–3410. Association for Computational Linguistics.
- Haoyue Shi, Jiayuan Mao, Kevin Gimpel, and Karen Livescu. 2019. [Visually grounded neural syntax acquisition](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1842–1861. Association for Computational Linguistics.

- Zhongmin Shi, Gabor Melli, Yang Wang, Yudong Liu, Baohua Gu, Mehdi M Kashani, Anoop Sarkar, and Fred Popowich. 2007. Question answering summarization of multiple biomedical documents. In *Advances in Artificial Intelligence*, pages 284–295. Springer.
- Hiroki Shimanaka, Tomoyuki Kajiwara, and Mamoru Komachi. 2018. [RUSE: regressor using sentence embeddings for automatic machine translation evaluation](#). In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers, WMT 2018, Belgium, Brussels, October 31 - November 1, 2018*, pages 751–758. Association for Computational Linguistics.
- Hoo-Chang Shin, Kirk Roberts, Le Lu, Dina Demner-Fushman, Jianhua Yao, and Ronald M. Summers. 2016. [Learning to read chest x-rays: Recurrent neural cascade model for automated image annotation](#). In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 2497–2506. IEEE Computer Society.
- Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. [Auto-prompt: Eliciting knowledge from language models with automatically generated prompts](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 4222–4235. Association for Computational Linguistics.
- Robik Shrestha, Kushal Kafle, and Christopher Kanan. 2020. [A negative case analysis of visual grounding methods for VQA](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 8172–8181. Association for Computational Linguistics.
- Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Motlaghi, Luke Zettlemoyer, and Dieter Fox. 2020. [ALFRED: A benchmark for interpreting grounded instructions for everyday tasks](#). In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 10737–10746. Computer Vision Foundation / IEEE.
- Kurt Shuster, Samuel Humeau, Antoine Bordes, and Jason Weston. 2018. Engaging image chat: Modeling personality in grounded dialogue. *arXiv preprint arXiv:1811.00945*.
- Kurt Shuster, Samuel Humeau, Antoine Bordes, and Jason Weston. 2020. [Image-chat: Engaging grounded conversations](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 2414–2429. Association for Computational Linguistics.
- Ekaterina Shutova, Niket Tandon, and Gerard de Melo. 2015. [Perceptually grounded selectional preferences](#). In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 950–960. The Association for Computer Linguistics.
- Vered Shwartz, Rachel Rudinger, and Oyvind Tafjord. 2020a. ["you are grounded!": Latent name artifacts in pre-trained language models](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6850–6861. Association for Computational Linguistics.

- Vered Shwartz, Peter West, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020b. [Un-supervised commonsense question answering with self-talk](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 4615–4629. Association for Computational Linguistics.
- Carina Silberer and Mirella Lapata. 2014. [Learning grounded meaning representations with autoencoders](#). In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers*, pages 721–732. The Association for Computer Linguistics.
- Carina Silberer and Manfred Pinkal. 2018. [Grounding semantic roles in images](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2616–2626. Association for Computational Linguistics.
- R Mahesh K Sinha and Anil Thakur. 2005. Machine translation of bi-lingual hindi-english (hinglish) text. *10th Machine Translation summit (MT Summit X), Phuket, Thailand*, pages 149–156.
- Sunayana Sitaram and Alan W Black. 2016. Speech Synthesis of Code-Mixed Text. In *LREC*.
- Sunayana Sitaram, Khyathi Raghavi Chandu, Sai Krishna Rallabandi, and Alan W Black. 2019. A survey of code-switched speech and language processing. *arXiv preprint arXiv:1904.00784*.
- Sunayana Sitaram, Sai Krishna Rallabandi, and Shruti Rijhwani1 Alan W Black. Experiments with Cross-lingual Systems for Synthesis of Code-Mixed Text. In *9th ISCA Speech Synthesis Workshop*, pages 76–81.
- Richard Skiba. 1997. Code switching as a countenance of language interference. *The internet TESL journal*, 3(10):1–6.
- Jonathan Slocum. 1985. A survey of machine translation: its history, current status, and future prospects. *Computational linguistics*, 11(1):1–17.
- Marko Smilevski, Ilija Lalkovski, and Gjorgji Madzarov. 2018. Stories for images-in-sequence by using visual and narrative components. *arXiv preprint arXiv:1805.05622*.
- Tamar Solorio and Yang Liu. 2008a. Learning to predict code-switching points. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 973–981. Association for Computational Linguistics.
- Tamar Solorio and Yang Liu. 2008b. Part-of-speech tagging for english-spanish code-switched text. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 1051–1060.
- Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. Mass: Masked sequence to sequence pre-training for language generation. *arXiv preprint arXiv:1905.02450*.
- Zhiyi Song, Stephanie M Strassel, Haejoong Lee, Kevin Walker, Jonathan Wright, Jennifer Garland, Dana Fore, Brian Gainor, Preston Cabe, Thomas Thomas, et al. 2014. Collecting natural sms and chat conversations in multiple languages: The bolt phase 2 corpus. In *LREC*, pages 1699–1704. Citeseer.

- Georgios P. Spithourakis, Isabelle Augenstein, and Sebastian Riedel. 2016. **Numerically grounded language models for semantic error correction**. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 987–992. The Association for Computational Linguistics.
- Tejas Srinivasan, Ramon Sanabria, Florian Metze, and Desmond Elliott. 2020. **Fine-grained grounding for multimodal speech recognition**. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020, Online Event, 16-20 November 2020*, pages 2667–2677. Association for Computational Linguistics.
- Milos Stanojevic and Khalil Sima'an. 2014. **BEER: better evaluation as ranking**. In *Proceedings of the Ninth Workshop on Statistical Machine Translation, WMT@ACL 2014, June 26-27, 2014, Baltimore, Maryland, USA*, pages 414–419. The Association for Computer Linguistics.
- Michael Q Stearns, Colin Price, Kent A Spackman, and Amy Y Wang. 2001. Snomed clinical terms: overview of the development process and project status. In *Proceedings of the AMIA Symposium*, page 662. American Medical Informatics Association.
- Luc Steels. 2004. **Constructivist development of grounded construction grammar**. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics, 21-26 July, 2004, Barcelona, Spain*, pages 9–16. ACL.
- Jonathan Stray. 2016. What do journalists do with documents? field notes for natural language processing researchers. In *Computation & Journalism Symposium. Palo Alto, CA: Stanford University*. [https://journalism.stanford.edu/cj2016/files/What do journalists do with documents.pdf](https://journalism.stanford.edu/cj2016/files/What%20do%20journalists%20do%20with%20documents.pdf).
- Jonathan Stray. 2019. Making artificial intelligence work for investigative journalism. *Digital Journalism*, 7(8):1076–1097.
- Michael Strube and Udo Hahn. 1999. Functional centering - grounding referential coherence in information structure. *Comput. Linguistics*, 25(3):309–344.
- Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2020. VL-BERT: Pre-training of generic visual-linguistic representations. In *ICLR*.
- Sanjay Subramanian, Lucy Lu Wang, Ben Bogin, Sachin Mehta, Madeleine van Zuylen, Sravanthi Parasa, Sameer Singh, Matt Gardner, and Hannaneh Hajishirzi. 2020. **Medicat: A dataset of medical images, captions, and textual references**. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 2112–2120. Association for Computational Linguistics.
- Akhilesh Sudhakar, Bhargav Upadhyay, and Arjun Maheswaran. 2019. **“transforming” delete, retrieve, generate approach for controlled text style transfer**. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3267–3277. Association for Computational Linguistics.

- Alessandro Suglia, Ioannis Konstas, Andrea Vanzo, Emanuele Bastianelli, Desmond Elliott, Stella Frank, and Oliver Lemon. 2020. [Compguesswhat?!: A multi-task evaluation framework for grounded language learning](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7625–7641. Association for Computational Linguistics.
- Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. 2019. [A corpus for reasoning about natural language grounded in photographs](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 6418–6428. Association for Computational Linguistics.
- Jiamei Sun, Sebastian Lapuschkin, Wojciech Samek, and Alexander Binder. 2022. [Explain and improve: Lrp-inference fine-tuning for image captioning models](#). *Inf. Fusion*, 77:233–246.
- Ece Takmaz, Mario Giulianelli, Sandro Pezzelle, Arabella Sinclair, and Raquel Fernández. 2020. Refer, reuse, reduce: Grounding subsequent references in visual and conversational contexts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4350–4368.
- Yik-Cheung Tam. 2020. Cluster-based beam search for pointer-generator chatbot grounded by knowledge. *Computer Speech & Language*, page 101094.
- Pradyumna Tambwekar, Murtaza Dhuliawala, Animesh Mehta, Lara J. Martin, Brent Harrison, and Mark O. Riedl. 2018. [Controllable neural story generation via reinforcement learning](#). *CoRR*, abs/1809.10736.
- Bowen Tan, Zichao Yang, Maruan Al-Shedivat, Eric P. Xing, and Zhiting Hu. 2020. [Progressive generation of long text](#). *CoRR*, abs/2006.15720.
- Hao Tan and Mohit Bansal. 2019. LXMERT: Learning cross-modality encoder representations from transformers. In *EMNLP-IJCNLP*.
- Yi Chern Tan and L. Elisa Celis. 2019. [Assessing social and intersectional biases in contextualized word representations](#). In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 13209–13220.
- Ying Hua Tan and Chee Seng Chan. 2016. [phi-lstm: A phrase-based hierarchical LSTM model for image captioning](#). In *Computer Vision - ACCV 2016 - 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part V*, volume 10115 of *Lecture Notes in Computer Science*, pages 101–117. Springer.
- Oguzhan Tas and Farzad Kiyani. A survey automatic text summarization. *PressAcademia Procedia*, 5(1):205–213.
- Wilson L Taylor. 1953. “cloze procedure”: A new tool for measuring readability. *Journalism quarterly*, 30(4):415–433.

- Ashish V. Thapliyal and Radu Soricut. 2020. [Cross-modal language generation using pivot stabilization for web-scale language coverage](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 160–170. Association for Computational Linguistics.
- Lucas Theis, Aäron van den Oord, and Matthias Bethge. 2016. [A note on the evaluation of generative models](#). In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*.
- Jesse Thomason, Michael Murray, Maya Cakmak, and Luke Zettlemoyer. 2019. [Vision-and-dialog navigation](#). In *3rd Annual Conference on Robot Learning, CoRL 2019, Osaka, Japan, October 30 - November 1, 2019, Proceedings*, volume 100 of *Proceedings of Machine Learning Research*, pages 394–406. PMLR.
- James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2019. [Generating token-level explanations for natural language inference](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 963–969. Association for Computational Linguistics.
- Liang Tian, Derek F Wong, Lidia S Chao, Paulo Quaresma, Francisco Oliveira, and Lu Yi. 2014. Um-corpus: A large english-chinese parallel corpus for statistical machine translation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*.
- Ran Tian, Shashi Narayan, Thibault Sellam, and Ankur P. Parikh. 2019. [Sticking to the facts: Confident decoding for faithful data-to-text generation](#). *CoRR*, abs/1910.08684.
- Julian Togelius, Georgios N. Yannakakis, Kenneth O. Stanley, and Cameron Browne. 2011. [Search-based procedural content generation: A taxonomy and survey](#). *IEEE Trans. Comput. Intell. AI Games*, 3(3):172–186.
- Chao Tong, Richard C. Roberts, Rita Borgo, Sean P. Walton, Robert S. Laramée, Kodzo Wegba, Aidong Lu, Yun Wang, Huamin Qu, Qiong Luo, and Xiaojuan Ma. 2018. [Storytelling and visualization: An extended survey](#). *Information*, 9(3):65.
- Christof Traber, Karl Huber, Karim Nedir, Beat Pfister, Eric Keller, and Brigitte Zellner. 1999. From multilingual to polyglot speech synthesis. In *Eurospeech*, pages 835–838.
- Chen-Tse Tsai and Dan Roth. 2016a. [Concept grounding to multiple knowledge bases via indirect supervision](#). *Trans. Assoc. Comput. Linguistics*, 4:141–154.
- Chen-Tse Tsai and Dan Roth. 2016b. [Illinois cross-lingual wikifier: Grounding entities in many languages to the english wikipedia](#). In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference System Demonstrations, December 11-16, 2016, Osaka, Japan*, pages 146–150. ACL.
- George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, et al. 2015. An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. *BMC bioinformatics*, 16(1):138.

- Masatoshi Tsuchiya. 2018. [Performance impact caused by hidden bias of training data for recognizing textual entailment](#). In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018*. European Language Resources Association (ELRA).
- Satoshi Tsutsui and David Crandall. 2017. Using artificial tokens to control languages for multilingual image caption generation. *arXiv preprint arXiv:1706.06275*.
- Selen Türkay. 2016. The effects of whiteboard animations on retention and subjective experiences when learning advanced physics topics. *Computers & Education*, 98:102–114.
- Takuma Udagawa, Takato Yamazaki, and Akiko Aizawa. 2020. [A linguistic analysis of visually grounded dialogues based on spatial expressions](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020, Online Event, 16-20 November 2020*, pages 750–765. Association for Computational Linguistics.
- Ted Underwood, David Bamman, and Sabrina Lee. The transformation of gender in english-language fiction. *Journal ISSN*, 2371:4549.
- Len Unsworth. 2008. *Multimodal semiotics: Functional analysis in contexts of education*. Bloomsbury Publishing.
- Oleg V. Vasilyev, Vedant Dharnidharka, and John Bohannon. 2020. [Fill in the BLANC: human-free quality estimation of document summaries](#). *CoRR*, abs/2002.09836.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017a. [Attention is all you need](#). In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 5998–6008.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017b. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.
- Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2015. [Cider: Consensus-based image description evaluation](#). In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 4566–4575. IEEE Computer Society.
- Subhashini Venugopalan, Lisa Anne Hendricks, Marcus Rohrbach, Raymond J. Mooney, Trevor Darrell, and Kate Saenko. 2017. [Captioning images with diverse objects](#). In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 1170–1178. IEEE Computer Society.
- Ashwin K Vijayakumar, Michael Cogswell, Ramprasath R Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. 2016. Diverse beam search: Decoding diverse solutions from neural sequence models. *arXiv preprint arXiv:1610.02424*.
- Ashwin K. Vijayakumar, Ramakrishna Vedantam, and Devi Parikh. 2017. [Sound-word2vec: Learning word representations grounded in sounds](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 920–925. Association for Computational Linguistics.

- Harm de Vries, Kurt Shuster, Dhruv Batra, Devi Parikh, Jason Weston, and Douwe Kiela. 2018. Talk the walk: Navigating new york city through grounded dialogue. *arXiv preprint arXiv:1807.03367*.
- Harm de Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron C. Courville. 2017. [Guesswhat?! visual object discovery through multi-modal dialogue](#). In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 4466–4475. IEEE Computer Society.
- Hoa Trong Vu, Claudio Greco, Aliia Erofeeva, Somayeh Jafaritazehjan, Guido Linders, Marc Tanti, Alberto Testoni, Raffaella Bernardi, and Albert Gatt. 2018. [Grounded textual entailment](#). In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 2354–2368. Association for Computational Linguistics.
- Yogarshi Vyas, Spandana Gella, Jatin Sharma, Kalika Bali, and Monojit Choudhury. 2014. Post tagging of english-hindi code-mixed social media content. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 974–979.
- Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020a. Asking and answering questions to evaluate the factual consistency of summaries. *arXiv preprint arXiv:2004.04228*.
- Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020b. [Asking and answering questions to evaluate the factual consistency of summaries](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 5008–5020. Association for Computational Linguistics.
- Cunxiang Wang, Pai Liu, and Yue Zhang. 2021a. [Can generative pre-trained language models serve as knowledge bases for closed-book qa?](#) In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 3241–3251. Association for Computational Linguistics.
- Dalin Wang, Daniel Beck, and Trevor Cohn. 2019a. On the role of scene graphs in image captioning. In *Proceedings of the Beyond Vision and LANguage: inTEgrating Real-world kNowledge (LANTERN)*, pages 29–34.
- Hao Wang, Doyen Sahoo, Chenghao Liu, Ee-Peng Lim, and Steven C. H. Hoi. 2019b. [Learning cross-modal embeddings with adversarial networks for cooking recipes and food images](#). In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 11572–11581. Computer Vision Foundation / IEEE.
- Peng Wang, Junyang Lin, An Yang, Chang Zhou, Yichang Zhang, Jingren Zhou, and Hongxia Yang. 2021b. [Sketch and refine: Towards faithful and informative table-to-text generation](#). In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 4831–4843. Association for Computational Linguistics.
- Pidong Wang and Hwee Tou Ng. 2013. A beam-search decoder for normalization of social media text with application to machine translation. In *Proceedings of the 2013 Conference of the*

- North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 471–481.
- Qinxin Wang, Hao Tan, Sheng Shen, Michael W. Mahoney, and Zhewei Yao. 2020c. [MAF: multimodal alignment framework for weakly-supervised phrase grounding](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 2030–2038. Association for Computational Linguistics.
- Tianlu Wang, Jieyu Zhao, Mark Yatskar, Kai-Wei Chang, and Vicente Ordonez. 2019c. Balanced datasets are not enough: Estimating and mitigating gender bias in deep image representations. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5310–5319.
- Wenlin Wang, Zhe Gan, Hongteng Xu, Ruiyi Zhang, Guoyin Wang, Dinghan Shen, Changyou Chen, and Lawrence Carin. 2019d. Topic-guided variational auto-encoder for text generation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 166–177.
- William Yang Wang and Miaomiao Wen. 2015. I can has cheezburger? a nonparanormal approach to combining textual and visual information for predicting and generating popular meme descriptions. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 355–365.
- Xin Wang, Wenhui Chen, Yuan-Fang Wang, and William Yang Wang. 2018. No metrics are perfect: Adversarial reward learning for visual storytelling. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 899–909.
- Xuancong Wang, Hwee Tou Ng, and Khe Chai Sim. 2014. A beam-search decoder for disfluency detection. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1457–1467.
- Yiren Wang, Fei Tian, Di He, Tao Qin, ChengXiang Zhai, and Tie-Yan Liu. 2019e. [Non-autoregressive machine translation with auxiliary regularization](#). In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 5377–5384. AAAI Press.
- Yufei Wang, Zhe Lin, Xiaohui Shen, Scott Cohen, and Garrison W Cottrell. 2017. Skeleton key: Image captioning by skeleton-attribute decomposition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7272–7281.
- Kellie Webster, Marta Recasens, Vera Axelrod, and Jason Baldridge. 2018. [Mind the GAP: A balanced corpus of gendered ambiguous pronouns](#). *Trans. Assoc. Comput. Linguistics*, 6:605–617.
- Dirk Weissenborn, George Tsatsaronis, and Michael Schroeder. 2013. Answering factoid questions in the biomedical domain. *BioASQ@ CLEF*, 1094.

- Sean Welleck, Ilya Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. 2020. [Neural text generation with unlikelihood training](#). In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.
- Sarah Wiegrefe and Yuval Pinter. 2019. [Attention is not not explanation](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 11–20. Association for Computational Linguistics.
- Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256.
- Genta Indra Winata, Andrea Madotto, Chien-Sheng Wu, and Pascale Fung. 2019. Code-switched language models using neural based synthetic data from parallel sentences. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 271–280.
- Sam Wiseman, Stuart Shieber, and Alexander Rush. 2018. Learning neural templates for text generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3174–3187.
- Yuk Wah Wong and Raymond Mooney. 2007. Generation by inverting a semantic parser that uses statistical machine translation. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 172–179.
- Bowen Wu, Mengyuan Li, Zongsheng Wang, Yifu Chen, Derek F. Wong, Qihang Feng, Junhong Huang, and Baoxun Wang. 2020. [Guiding variational response generator to exploit persona](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 53–65. Association for Computational Linguistics.
- Qi Wu, Damien Teney, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. 2017. Visual question answering: A survey of methods and datasets. *Computer Vision and Image Understanding*, 163:21–40.
- Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016a. [Google’s neural machine translation system: Bridging the gap between human and machine translation](#). *CoRR*, abs/1609.08144.
- Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016b. Google’s neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

- Chen Xing, Wei Wu, Yu Wu, Jie Liu, Yalou Huang, Ming Zhou, and Wei-Ying Ma. 2017. [Topic aware neural response generation](#). In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 3351–3357. AAAI Press.
- Jingjing Xu, Xuancheng Ren, Yi Zhang, Qi Zeng, Xiaoyan Cai, and Xu Sun. 2018a. A skeleton-based model for promoting coherence among sentences in narrative story generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4306–4315.
- Jingjing Xu, Xu Sun, Xuancheng Ren, Junyang Lin, Bingzhen Wei, and Wei Li. 2018b. [DP-GAN: diversity-promoting generative adversarial network for generating informative and diversified text](#). *CoRR*, abs/1802.01345.
- Jun Xu, Haifeng Wang, Zheng-Yu Niu, Hua Wu, Wanxiang Che, and Ting Liu. 2020. [Conversational graph grounded policy learning for open-domain conversation generation](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 1835–1845. Association for Computational Linguistics.
- Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057.
- Peng Xu, Yanshuai Cao, and Jackie Chi Kit Cheung. 2019. Unsupervised controllable text generation with global variation discovery and disentanglement. *arXiv preprint arXiv:1905.11975*.
- Marina Yaguello et al. 1998. *Language through the looking glass: Exploring language and linguistics*. Oxford University Press on Demand.
- Min Yang, Wei Zhao, Wei Xu, Yabing Feng, Zhou Zhao, Xiaojun Chen, and Kai Lei. 2019a. [Multitask learning for cross-domain image captioning](#). *IEEE Trans. Multim.*, 21(4):1047–1061.
- Pengcheng Yang, Fuli Luo, Peng Chen, Lei Li, Zhiyi Yin, Xiaodong He, and Xu Sun. 2019b. Knowledgeable storyteller: a commonsense-driven generative model for visual storytelling. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 5356–5362. AAAI Press.
- Shaohua Yang, Qiaozhi Gao, Changsong Liu, Caiming Xiong, Song-Chun Zhu, and Joyce Y. Chai. 2016. [Grounded semantic role labeling](#). In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 149–159. The Association for Computational Linguistics.
- Tsung-Yen Yang, Andrew S. Lan, and Karthik Narasimhan. 2020. [Robust and interpretable grounding of spatial references with relation networks](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020, Online Event, 16-20 November 2020*, pages 1908–1923. Association for Computational Linguistics.
- Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. 2019c. Auto-encoding scene graphs for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10685–10694.

- Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019d. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5754–5764.
- Ziyi Yang, Chenguang Zhu, Vin Sachidananda, and Eric Darve. 2019e. [Embedding imputation with grounded language information](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3356–3361. Association for Computational Linguistics.
- Lili Yao, Nanyun Peng, Weischedel Ralph, Kevin Knight, Dongyan Zhao, and Rui Yan. 2018a. Plan-and-write: Towards better automatic storytelling. *arXiv preprint arXiv:1811.05701*.
- Lili Yao, Nanyun Peng, Ralph Weischedel, Kevin Knight, Dongyan Zhao, and Rui Yan. 2019. Plan-and-write: Towards better automatic storytelling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7378–7385.
- Shaowei Yao and Xiaojun Wan. 2020. [Multimodal transformer for multimodal machine translation](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 4346–4350. Association for Computational Linguistics.
- Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. 2018b. Exploring visual relationship for image captioning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 684–699.
- Ting Yao, Yingwei Pan, Yehao Li, Zhaofan Qiu, and Tao Mei. 2017. Boosting image captioning with attributes. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4894–4902.
- Mahsa Yarmohammadi, Vivek Kumar Rangarajan Sridhar, Srinivas Bangalore, and Baskaran Sankaran. 2013. Incremental segmentation and decoding strategies for simultaneous translation. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1032–1036.
- Mark Yatskar, Bo Pang, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. 2010. [For the sake of simplicity: Unsupervised extraction of lexical simplifications from wikipedia](#). In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 2-4, 2010, Los Angeles, California, USA*, pages 365–368. The Association for Computational Linguistics.
- Harish Yenala, Avinash Kamineni, Manish Shrivastava, and Manoj Kumar Chinnakotla. 2015. Iiith at bioasq challenge 2015 task 3b: Bio-medical question answering system. In *CLEF (Working Notes)*.
- Da Yin, Liunian Harold Li, Ziniu Hu, Nanyun Peng, and Kai-Wei Chang. 2021. [Broaden the vision: Geo-diverse visual commonsense reasoning](#). *CoRR*, abs/2109.06860.
- Yuya Yoshikawa, Yutaro Shigeto, and Akikazu Takeuchi. 2017. [STAIR captions: Constructing a large-scale japanese image caption dataset](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pages 417–421. Association for Computational Linguistics.

- Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. 2016. Image captioning with semantic attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4651–4659.
- Jun Yu, Jing Li, Zhou Yu, and Qingming Huang. 2019. Multimodal transformer with multi-view visual representation for image captioning. *arXiv*, 1905.07841.
- Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017. [Seqgan: Sequence generative adversarial nets with policy gradient](#). In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 2852–2858. AAAI Press.
- Sina Zarrieß and David Schlangen. 2017. [Deriving continuous grounded meaning representations from referentially structured multimodal contexts](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 959–965. Association for Computational Linguistics.
- Sina Zarrieß and David Schlangen. 2018. Decoding strategies for neural referring expression generation. *Proceedings of INLG 2018*.
- Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. [From recognition to cognition: Visual commonsense reasoning](#). In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 6720–6731. Computer Vision Foundation / IEEE.
- Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. [SWAG: A large-scale adversarial dataset for grounded commonsense inference](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 93–104. Association for Computational Linguistics.
- Fangzhou Zhai, Vera Demberg, Pavel Shkadzko, Wei Shi, and Asad Sayeed. 2019. A hybrid model for globally coherent story generation. In *Proceedings of the Second Workshop on Storytelling*, pages 34–45.
- Houyu Zhang, Zhenghao Liu, Chenyan Xiong, and Zhiyuan Liu. 2020a. [Grounded conversation generation as guided traverses in commonsense knowledge graphs](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 2031–2043. Association for Computational Linguistics.
- Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. 2021. [Vinvl: Revisiting visual representations in vision-language models](#). In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 5579–5588. Computer Vision Foundation / IEEE.
- Renxian Zhang. 2011. Sentence ordering driven by local and global coherence for summary generation. In *Proceedings of the ACL 2011 Student Session*, pages 6–11. Association for Computational Linguistics.
- Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? *arXiv preprint arXiv:1801.07243*.

- Shiyue Zhang and Mohit Bansal. 2021. [Finding a balanced degree of automation for summary evaluation](#). *CoRR*, abs/2109.11503.
- Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020b. [Bertscore: Evaluating text generation with BERT](#). In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.
- Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657.
- Yizhe Zhang, Zhe Gan, Kai Fan, Zhi Chen, Ricardo Henao, Dinghan Shen, and Lawrence Carin. 2017. Adversarial feature matching for text generation. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 4006–4015. JMLR. org.
- Tiancheng Zhao, Ran Zhao, and Maxine Eskénazi. 2017. [Learning discourse-level diversity for neural dialog models using conditional variational autoencoders](#). In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 654–664. Association for Computational Linguistics.
- Yanpeng Zhao and Ivan Titov. 2020. [Visually grounded compound pcfgs](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 4369–4379. Association for Computational Linguistics.
- Yue Zheng, Yali Li, and Shengjin Wang. 2019. [Intention oriented image captions with guiding objects](#). In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 8395–8404. Computer Vision Foundation / IEEE.
- Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020a. [How does NLP benefit legal system: A summary of legal artificial intelligence](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 5218–5230. Association for Computational Linguistics.
- Victor Zhong, Mike Lewis, Sida I. Wang, and Luke Zettlemoyer. 2020b. [Grounded adaptation for zero-shot executable semantic parsing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6869–6882. Association for Computational Linguistics.
- Ben Zhou, Daniel Khashabi, Chen-Tse Tsai, and Dan Roth. 2018a. [Zero-shot open entity typing as type-compatible grounding](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2065–2076. Association for Computational Linguistics.
- Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. 2018b. [Emotional chatting machine: Emotional conversation generation with internal and external memory](#). In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 730–739. AAAI Press.

- Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. 2018c. Emotional chatting machine: Emotional conversation generation with internal and external memory. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Kangyan Zhou, Shrimai Prabhunoye, and Alan W. Black. 2018d. [A dataset for document grounded conversations](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 708–713. Association for Computational Linguistics.
- Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason J. Corso, and Jianfeng Gao. 2020a. [Unified vision-language pre-training for image captioning and VQA](#). In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 13041–13049. AAAI Press.
- Luowei Zhou, Chenliang Xu, and Jason J Corso. 2018e. Towards automatic learning of procedures from web instructional videos. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Mingyang Zhou, Runxiang Cheng, Yong Jae Lee, and Zhou Yu. 2018f. [A visual attention grounding neural model for multimodal machine translation](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 3643–3653. Association for Computational Linguistics.
- Pei Zhou, Weijia Shi, Jieyu Zhao, Kuan-Hao Huang, Muhao Chen, Ryan Cotterell, and Kai-Wei Chang. 2019. [Examining gender bias in languages with grammatical gender](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 5275–5283. Association for Computational Linguistics.
- Yuanen Zhou, Meng Wang, Daqing Liu, Zhenzhen Hu, and Hanwang Zhang. 2020b. [More grounded image captioning by distilling image-text matching model](#). In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 4776–4785. Computer Vision Foundation / IEEE.
- Wanrong Zhu, Zhiting Hu, and Eric Xing. 2019. Text infilling. *arXiv preprint arXiv:1901.00158*.
- Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. [Texygen: A benchmarking platform for text generation models](#). In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, pages 1097–1100. ACM.