Coping With Ambiguity in a Large-Scale Machine Translation System

Kathryn L. Baker, Alexander M. Franz, Pamela W. Jordan,
Teruko Mitamura, Eric H. Nyberg, 3rd
Center for Machine Translation
Carnegie Mellon University
Pittsburgh, PA 15213

Topical Paper: machine translation, parsing

Abstract

In an interlingual knowledge-based machine translation system, ambiguity arises when the source language analyzer produces more than one interlingua expression for a source sentence. This can have a negative impact on translation quality, since a target sentence may be produced from an unintended meaning. In this paper we describe the methods used in the KANT machine translation system to reduce or eliminate ambiguity in a large-scale application domain. We also test these methods on a large corpus of test sentences, in order to illustrate how the different disambiguation methods reduce the average number of parses per sentence.

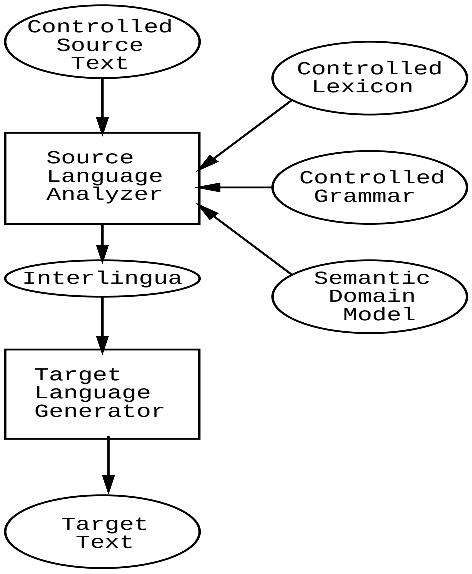

Figure 1: **The KANT System**

1 Introduction

The KANT system [Mitamura et al., 1991] is a system for Knowledge-based, Accurate Natural-language Translation. The system is used in focused technical domains for multilingual translation of controlled source language documents. KANT is an interlingua-based system: the source language analyzer produces an interlingua expression for each source sentence, and this interlingua is processed to produce the corresponding target sentence. The problem of *ambiguity* arises when the system produces more than one interlingua representation for a single input sentence. If the goal is to automate translation and produce output that does not require post-editing, then the presence of ambiguity has a negative impact on translation quality, since a target sentence may be produced from an unintended meaning. When it is possible to limit the interpretations of a sentence to just those that are coherent in the translation domain, then the accuracy of the MT system is enhanced.

Ambiguity can occur at different levels of processing in source analysis. In this paper, we describe how we cope with ambiguity in the KANT controlled lexicon, grammar, and semantic domain model, and how these are designed to reduce or eliminate ambiguity in a given translation domain.

2 Constraining the Source Text

The KANT domain lexicon and grammar are a constrained subset of the general source language lexicon and grammar. The strategy of constraining the source text has three main goals. First, it encourages clear and direct writing, which is beneficial to both the reader of the source text and to the translation process. Second, it facilitates consistent writing among the many authors who use the system and across all document types. And third, the selection of unambiguous words and constructions to be used during authoring reduces the necessity for ambiguity resolution during the automatic stages of processing. It is important to reduce the processing overhead associated with ambiguity resolution in order to keep the system fast enough for on-line use.

2.1 The Domain Lexicon

The domain lexicon is built using corpus analysis. Lists of terms, arranged by part of speech, are automatically extracted from the corpus [Mitamura et al., 1993]. The lexicon consists of *closed-class general words, open-class general words, idioms*, and *nomenclature phrases*. Closed-class general words (e.g. *the, with, should*) are taken from general English. Open-class general words (e.g. *drain, run, hot*) are limited in the lexicon to one sense per part of speech with some exceptions[1]. Idioms (e.g. *on and off*) and nomenclature phrases (e.g. *summing valve*) are domain-specific and are limited to those phrases identified in the domain corpus. Phrases, too, are defined with a single sense. Special vocab-

[1]For example, in the heavy-equipment lexicon, there are a few hundred terms out of 60,000 which have more than one sense per part of speech.

ulary items, including symbols, abbreviations, and the like, are restricted in use and are chosen for the lexicon in collaboration with domain experts. Senses for prepositions, which are highly ambiguous and context-dependent, are determined during processing using the semantic domain model (cf. Section 4).

Nominal compounds in the domain may be several words long. Because of the potential ambiguity associated with compositional parsing of nominal compounds, non-productive nominal compounds are listed explicitly in the lexicon as idioms or nomenclature phrases.

2.2 Controlled Grammar

Some constructions in the general source language that are inherently ambiguous are excluded from the restricted grammar, since they may lead to multiple analyses during processing:

- Conjunction of VPs, ADJs, or ADVs e.g. **Extend and retract the cylinder.*
- Pronominal reference, e.g. **Start the engine and keep it running.*
- Ellipsis, e.g. reduced relative clauses: **the tools used for the procedure*
- Long-distance dependencies, such as interrogatives and object-gap relative clauses, e.g. *The parts which the service representative ordered.*
- Nominal compounding which is not explicitly coded in the phrasal lexicon.

On the other hand, the grammar includes the following constructions:

- Active, passive and imperative sentences, e.g. *Start the engine.*
- Conjunction of NPs, PPs or Ss. Sentences may be conjoined using coordinate or subordinate conjunctions, e.g. *If you are on the last parameter, then the program proceeds to the top.*
- Subject-gap relative clauses, e.g. *The service representative can determine the parts which are faulty.*

The recommendations in the controlled grammar include guidelines for authoring, such as how to rewrite a text from general English into the domain language. Authors are advised, for example, to choose the most concise terms available in the lexicon and to rewrite long, conjoined sentences into short, simple ones. The recommendations are useful both for rewriting old text and creating new text (see Figure 2 for examples).

Example 1: **Rewrite Anaphoric Use of Numerals**

Problematic Text:	*Loosen the smaller one first.*
Suggested Rewrite:	*Loosen the smaller bolt first.*

Example 2: **Use Concise Vocabulary**

Problematic Text:	*The parts must be put back together.*
Suggested Rewrite:	*The parts must be reassembled.*

Figure 2: **Grammar Recommendation Examples**

2.3 SGML Text Markup

The grammar makes use of Standard Generalized Markup Language (SGML) text markup tags. The set of markup tags for our application were developed in conjunction with domain experts. A set of domain-specific tags is used not only to demarcate the text but also to identify the content of potentially ambiguous expressions, and to help during vocabulary checking. For example, at the lexical level, number tags identify numerals as diagram callouts, part numbers, product model numbers, or parts of measurement expressions. At the syntactic level, rules for tag combinations restrict how phrases may be constructed, as with tagged part numbers and part names (see Figure 3 for an example).

```
The <partno> 4S7527 </partno> <partname>
Hose Assembly </partname> <callout> 1
</callout> of the <partno> 5T6544
</partno> <partname> Brake Control Group
</partname> must now be connected to the
<partno> 4K2986 </partno> <partname>
Anchor Tee </partname>.
```

Figure 3: **Sample SGML Text Mark-Up**

3 Grammar Design Issues

The parser in KANT is based on the "Universal Parser" [Tomita and Carbonell, 1987]. The grammar consists of context-free rules that define the input's constituent structure (c-structure) and these rules are annotated with constraint equations that define the input's functional structure (f-structure). Tomita's parser compiles the grammar into an LR-table, and the constraint equations into Lisp code. Although this compilation results in fast run-time parsing, the need to minimize ambiguity still exists.

One source of ambiguity is the attachment site for a prepositional phrase. However, many of the PP attachments are encoded directly in the grammar because the syntactic context indicates an unambiguous attachment site. For example:

- A partitive where the PP attaches to the noun: *a gallon of antifreeze*.
- A pre-sentential PP where the PP attaches to the sentence: *For this test, ensure that a signal line is connected from the pump output to the pump compensator.*
- A PP attaches to the verb *be* when there is no predicate adjective: *The truck is in the shop*.
- A ditransitive verb where the PP attaches to the verb: *Give your suggestions to the dealer*.
- A stand-alone PP inside an SGML tag such as QUALIFIER where the PP attaches to the MDLDESC tag contents: *Inspect <mdldesc> all track-type tractors <qualifier> with hydraulic heads </qualifier> </mdldesc>*.

3.1 Passive vs. Copular with Participial?

There are many adjectives in English that have the same form as an -ed participle. For example:

The radius is poorly formed. (adjective)
The calibration mode is enabled by moving the rocker switch. (participle)

To distinguish the adjectival from the participial form we have added two heuristics to the constraint rules of the grammar. The first is to use verb class mapping information. If the verb is classified as being more active than stative, then the passive reading is preferred. So, for example, an intransitive verb would indicate an adjectival reading:

The display is faded. (adjective)

The second heuristic uses the notion of "quasi-agents". There are several prepositions that can introduce "quasi-agents" [Quirk et al., 1972], such as: *about, at, over, to, with*. If the domain model indicates that the -ed verb is a possible attachment site for a prepositional phrase occurring in the sentence, then the passive reading is preferred.

These two heuristics are incorporated into the constraints of rules involving predicate adjectives. If the -ed form is classified as active, or if there is a PP in the sentence that can attach to the -ed verb form, then the adjectival reading is ruled out. In the constraints of rules for the passive, the passive reading is ruled out if the -ed form is classified as stative.

3.2 Adverb or Adjective?

For the most part, each word in the system is limited to one meaning per part of speech. So while we have nearly eliminated one source of lexical ambiguity, there is still the problem of ambiguity between the various parts of speech for a particular word. While ambiguity between, for example, a noun and a verb is usually resolved by the syntactic context, parts of speech that participate in similar contexts are still a problem. For example, the content of the SGML tag, POSITION, can be an adjective or adverb phrase and "as [<adj>|<adv>] as" can contain either an adjective or an adverb. This means that an input such as "as fast as" would have two analyses. We have found with our domain that the correct thing to do is to prefer the adverb reading. We put this preference directly into the constraints of rules involving adjectives for which the same context allows an adverb. If the word is also an adverb then the adjective rule will fail. This allows the adverb reading to be preferred.

4 Semantic Domain Model

We have implemented a practical method for integrating semantic rules into an LR parser. The resulting system combines the merits of a semantic domain model with the generality and wide coverage of syntactic parsing, and is fast and efficient enough to remain practical.

4.1 Interleaved vs. Sentence-final Constraints

Some previous knowledge-based natural language analysis systems have constructed the semantic representation for the sentence in tandem with syntactic parsing. In this scheme semantic constraints from the domain model filter out semantically ill-formed representations and kill the associated parsing path. Examples include ABSITY [Hirst, 1986] and KBMT-89 [Goodman and Nirenburg, 1991]. Other previous systems have delayed semantic interpretation and application of semantic well-formedness constraints until after the syntactic parse.

Both of these schemes entail performance problems. The solution to this problem lies in importing the right type and right amount of semantic information into syntactic parsing. In KANT, the relevant knowledge sources are reorganized into data structures that are optimized for ambiguity resolution during parsing.

4.2 Example of Attachment Ambiguity

The knowledge-based disambiguation scheme covers Prepositional Phrase attachment, Noun-Noun compounding, and Adjective-Noun attachment. The remainder of this section discusses examples involving PP-attachment. The syntactic grammar contains two rules that allow these attachments:

VP $\longleftarrow$ VP PP
NP $\longleftarrow$ NP PP

Consider the sentence *Measure the voltage with the voltmeter.* Syntactically, the PP *with the voltmeter* can modify either the verb *measure*, or the noun *voltage*.

4.3 Structure and Content of the Domain Model

We use knowledge about the domain to resolve ambiguities like PP-attachment. The domain model contains all of the semantic concepts in the domain. *Leaf concepts*, such as `*O-VOLTMETER`, correspond closely to linguistic expressions. The concepts are arranged in an inheritance hierarchy, and other concepts, such as `*O-MEASURING-DEVICE`, represent abstract concepts. The domain model is implemented as a hierarchy of concepts. Constraints on possible attributes of concepts, along with semantic constraints on the fillers, are inherited through this hierarchy. Figure 4 shows an example.

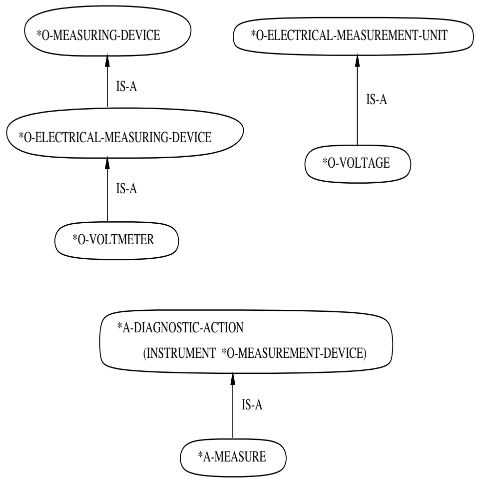

Figure 4: **Excerpt from Domain Model**

4.4 Using Semantics in the Syntax

In order to keep parsing tractable, the domain model is consulted at the earliest possible stage during parsing. Every grammar rule that involves an attachment decision that is subject to knowledge-based disambiguation calls a function that consults the domain model, and allows the grammar rule to succeed only if the attachment is semantically licensed. The grammar formalism allows procedural calls to be made directly from the grammar rules. The function that performs or denies attachment based on the domain model is called `sem-attach`.

The inputs to the `sem-attach` function are the functional structures (f-structures) for the potential attachment site, the structure to be attached, and the type of attachment (e.g., `PP` = Prepositional Phrase). `sem-attach` consults information from the domain model to decide whether the attachment is semantically licensed. This process is described in the next subsection.

4.5 Steps in Semantic Disambiguation

There are three main steps in semantic disambiguation of possible syntactic attachments: (1) mapping from syntax to semantic concepts using the lexical mapping rules; (2) checking information from the domain model; and (3) determining semantic roles using the semantic interpretation rules.

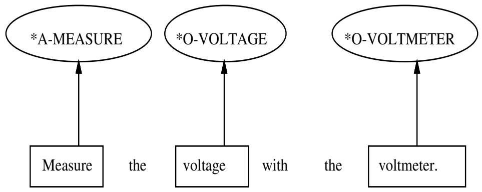

Figure 5: **Lexical Mapping Rules**

Lexical Mapping Rules. The first step is to map from syntactic structures to semantic concepts. The lexical mapping rules associate syntactic lexicon entries with concepts from the Domain Model (Figure 5).

Domain Model. The second step consists in looking up the appropriate concepts in the Domain Model (Figure 4).

Semantic Interpretation Rules. The third step consists of consulting the semantic interpretation rules to determine whether the concepts from the sentence can form appropriate modification relationships. Semantic interpretation rules describe the mapping from the syntactic representation to the frame-based semantic representation. An interpretation rule consists of a syntactic path (an index into the f-structure), a semantic path (an index into the semantic frame), and an optional syntactic constraint on the mapping rule. For example, below is an interpretation rule for the INSTRUMENT role:

```
(:syn-path (PP OBJ)
 :sem-path INSTRUMENT
 :syn-constraint
  ((pp ((root (*OR* "with" "by"))))))
```

Efficient Run-time Use. In order to make this process as efficient as possible, and to minimize delays during parsing, the knowledge described in this section is reorganized off-line before parsing. The result of this reorganization are data structures known as *semantic restrictors*. The semantic restrictors have three main properties:

1. They are indexed by head concept, and provide a list of all appropriate modifiers.
2. All inheritance in the Domain Model is performed off-line, so that the restrictors contain all necessary information.
3. The semantic restrictors are stored in a space-efficient structure-shared manner.

5 Author Disambiguation

Once KANT has analyzed a source sentence and all possible disambiguations have been performed, there may still be more than one interlingua representation for the sentence. This occurs when the sentence is truly ambiguous, i.e., it has more than one acceptable domain interpretation. In this case, KANT makes use of disambiguation by the author — the ambiguity is described to the author and the author is then prompted to select the desired interpretation. The choice is "remembered" by placing extra information into the input text at the point of ambiguity. There are two types of ambiguity currently addressed by author disambiguation:

- *Lexical Ambiguity*. When more than one interlingua is produced because a certain word or phrase can be interpreted in more than one way (ie. as two different concepts), then the author is prompted to select the desired meaning.
- *Structural Ambiguity*. When more than one attachment site is possible for a phrase like a prepositional phrase, the different attachments are glossed for the author, who is then prompted to select the desired interpretation.

Since author disambiguation is utilized only when the sentence cannot be disambiguated by other means, it will not occur very frequently once the system is complete. On the other hand, having such a mechanism available during system development is very helpful, since it helps to point out where there is residual ambiguity left to be addressed by knowledge source refinement.

6 Testing Disambiguation Methods

When disambiguation methods are introduced, the number of parses per sentence can be reduced dramatically. If we use a general lexicon and grammar to parse texts from a specialized domain corpus (rather than a general corpus), then more parses will be assigned than those that are desired in the domain. Figure 6 illustrates how the successive introduction of disambiguation methods reduces the set of possible parses to just those desired in the domain. The smallest set of interpretations is that remaining after the controlled lexicon, grammar, semantic restrictions, and author disambiguation have been applied; in practice this set will contain just one interpretation, since the author will select only the intended interpretation.

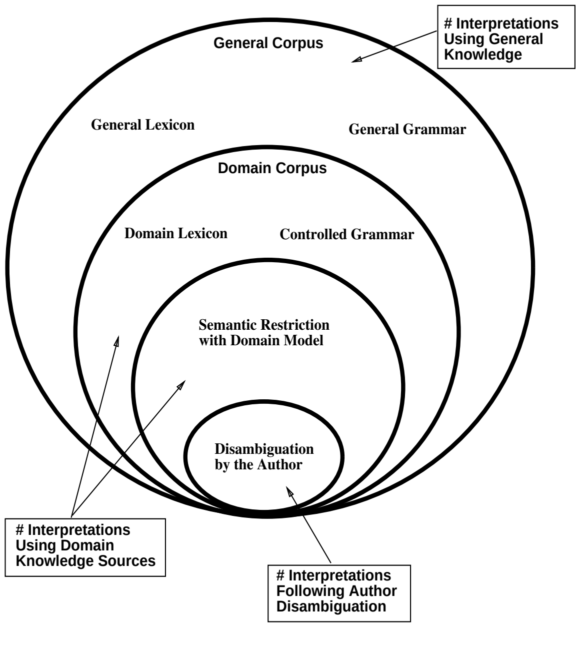

Figure 6: **Reducing the Set of Possible Interpretations**

We have experimented with the KANT analyzer in order to determine the effects of the different disambiguation strategies mentioned above. We used a test suite containing 891 sentences which is used for regression testing during system development. The sentences in the test suite range in length from 1 word to over 25 words.

General lexicon entries were derived automatically from the online version of Webster's 7th dictionary. Webster's includes 55,000 roots that are in at least one open class category (verb, noun, adjective, adverb). One dictionary entry was created for each sense of one of these categories. This resulted in 117,000 lexicon entries. The constrained lexicon consists of 10,000 words and 50,000 phrases tailored to the application domain. For the results listed below, the "general lexicon" consists of the constrained lexicon plus the general entries from Webster's.

The constrained grammar has been tailored to the restricted source language for the domain (cf. Section 2). In addition, it includes a number of constraint annotations and parse preferences that limit the number of ambiguous parses (cf. Section 3). A general grammar was derived from the constrained grammar by removing most restrictions and constraints on specific rules, leaving only the most general constraints such as subject-verb agreement.

When noun-noun compounding is allowed, sequences of nouns may form NPs even if they are not listed as nomenclature phrases in the lexicon. Each such sequence is only parsed one way; the parser does not build different structures for the sequence of nouns, but just reads them into a list.

In order to reduce the exponential complexity of some of the longer sentences, all test results were produced using the "shared packed forest" method of ambiguity packing for ambiguity internal to a sentence [Tomita, 1986]. The results for "parses per sentence" is simply the average for all the sentences.

Test	LEX	GRA	N-N	DM	P
1	GEN	GEN	YES	NO	27.0
2	GEN	GEN	NO	NO	10.2
3	GEN	CON	YES	NO	8.4
4	CON	GEN	YES	NO	1.7
5	CON	GEN	NO	NO	1.6
6	CON	CON	NO	YES	1.5

LEX: Lexicon GEN: General
GRA: Grammar CON: Constrained
N-N: Noun-Noun Compounding
DM: Semantic Restriction with Domain Model

Figure 7: **Testing Disambiguation Methods (12/17/93)**

The results of this testing are shown in Figure 7. Test 1 is the baseline result for parsing with a general lexicon, general grammar, noun-noun compounding and no semantic restrictions. As expected, the average number of parses per sentence is quite high (27.0). Limiting noun-noun compounding (Test 2) cuts this number by more than half, yielding 10.2 parses per sentence. Note that a similar effect is achieved if we run the test with a controlled grammar and noun-noun compounding (Test 3, 8.4 parses per sentence).

Constraining the lexicon seems to achieve the largest reduction in the average number of parses per sentence (Tests 4, 5, 6), with elimination of noun-noun compounding yielding only slight improvements when the lexicon has already been restricted. As expected, the best results are achieved when the system is run with constrained lexicon and grammar, no noun-noun compounding, and semantic restriction with a domain model (Test 6).

We expect that the primary reason why the addition of semantic restrictions from a domain model does not have a greater impact is due to the incomplete nature of the domain model we used in the experiment. The domain model used in the experiment captures the domain relationships associated with prepositional phrase attachment to VP and object NP, but there are several areas of the domain model still under development. When complete, these will further reduce ambiguity by placing additional limitations on the following:

- The semantic classification of words inside particular SGML tags;
- Attachment of prepositional phrases to subject NP;
- Attachment of infinitive clauses;
- Attachment of relative clauses.

This testing has proved extremely useful in prioritizing the level of effort expended on different disambiguation methods during system development. As is often the case, theoretically interesting or difficult issues (such as noun-noun compounding) are reduced in effect when other domain-related restrictions are put in place (such as a controlled lexicon). On the other hand, this type of testing can also identify the areas of the system (such as the semantic domain model) which are not reducing ambiguity as much as expected. In our ongoing work, we will complete the domain model for the KANT heavy-equipment application in those areas mentioned above; in the process, we expect to reduce the average number of parses per sentence in the most constrained case.

7 Acknowledgements

We would like to thank Jaime Carbonell, Radha Rao, and Todd Kaufmann, and all of our colleagues on the KANT project, including James Altucher, Nicholas Brownlow, Mildred Galarza, Sue Holm, Kathi Iannamico, Kevin Keck, Marion Kee, Sarah Law, John Leavitt, Daniela Lonsdale, Deryle Lonsdale, Jeanne Mier, Venkatesh Narayan, Amalio Nieto, and Will Walker, our sponsors at Caterpillar Inc., and our colleagues at Carnegie Group.

References

[Chevalier et al., 1978] Chevalier, M., Dansereau, J., and Poulin, G. (1978). Taum-meteo: description du système. Technical report, Groupe de recherches pour la traduction automatique, Université de Montrèal.

[Goodman and Nirenburg, 1991] Goodman, K. and Nirenburg, S. (1991). *The KBMT Project: A Case Study in Knowledge-Based Machine Translation*. Morgan Kaufmann, San Mateo, CA.

[Hirst, 1986] Hirst, G. (1986). *Semantic Interpretation and the Resolution of Ambiguity*. Cambridge University Press, Cambridge.

[Mitamura et al., 1991] Mitamura, T., Nyberg, E., and Carbonell, J. (1991). An efficient interlingua translation system for multi-lingual document production. In *Proceedings of Machine Translation Summit III*, Washington, DC.

[Mitamura et al., 1993] Mitamura, T., Nyberg, E., and Carbonell, J. (1993). Automated corpus analysis and the acquisition of large, multi-lingual knowledge bases for MT. In *5th International Conference on Theoretical and Methodological Issues in Machine Translation*, Kyoto, Japan.

[Quirk et al., 1972] Quirk, R., Greenbaum, S., Leech, G., and Svartvik, J. (1972). *A Grammar of Contemporary English*. Longman Group UK Limited, Essex England.

[Suzuki, 1992] Suzuki, M. (1992). A method of utilizing domain and language-specific constraints in dialog translation. In *Coling-92*.

[Tomita, 1986] Tomita, M. (1986). *Efficient Parsing for Natural Language*. Kluwer Academic Publishers, Boston, MA.

[Tomita and Carbonell, 1987] Tomita, M. and Carbonell, J. (1987). The Universal Parser architecure for Knowledge-based Machine Translation. Technical Report CMU-CMT-87-101, Center for Machine Translation, Carnegie Mellon University.