Translational Ambiguity Rephrased

Danit Ben-Ari*, Daniel M. Berry**, Mori Rimon*

*IBM Israel Scientific Center, Haifa, Israel
**Computer Science Department, Technion, Haifa, Israel

E-mail address: danit@israearn.bitnet

Abstract

Presented are the special aspects of translation-oriented disambiguation, which differentiate it from conventional text-understanding-oriented disambiguation. Also presented are the necessity of interaction to cover the failure of automatic disambiguation, and the idea of disambiguation by rephrasing. The types of ambiguities to which rephrasing is applicable are defined, and the four stages of the rephrasing procedure are described for each type of ambiguity. The concept of an interactive disambiguation module, which is logically located between the parser and the transfer phase, is described. The function of this module is to bridge the gap between several possible trees and/or other ambiguities, and one well-defined tree that may be satisfactorily translated.

1. Introduction

Natural languages are rich in ambiguities of many kinds *(see examples* 1, 3, 5 *and 1).* Furthermore, many sentences that do not seem ambiguous to humans, due to their extensive world knowledge, may present ambiguities to automatic parsers and to machine translation systems *(see examples 2,* 4, 6 *and* 8). Some of the more common sources of ambiguity are:

- Categorial ambiguity reflected in different part of speech (pos) assigned to words (in a way which yields valid parses):

  (1) The management **requests control** information.
  (2) **Visiting grandfather** will take up the whole day.

- Structural ambiguity where, although the words of the sentence are assigned identical pos, the sentence may have more than one valid parse:

  (3) Businessmen who are afraid to take risks **frequently** lose out to their competitors.
  (4) **Good** boys and girls go to heaven.

- Word sense ambiguity that is not reflected in parse trees but may be crucial to translation (both polysemy and homonymy):

  (5) Dan walked to the **bank.**
  (6) The thief stole two million dollars from the **bank.**

- Implicit ambiguity not explicitly demonstrated in parse representation, resulting from anaphora, control possibilities, gaps, and other hidden information that may be extracted in different ways.

  (7) The chicken are ready to **eat**.
  (8) The stone hit the shop window and it broke.

The issue of automatic disambiguation has been widely dealt with ([Birnbaum 85], [Milne 86], [Duffy 86], [Dahlgren 86], [Hirst 87], to mention just a few). Some of the problems may be handled syntactically e.g. by defining sub-categorization for verbs, assuming a heuristic procedure for PP attachment, etc.. In other cases, semantic devices such as selectional restrictions, frames, etc. are needed. Still there are cases in which the world knowledge needed for disambiguation is so large that it is difficult to conceive of a satisfactory solution by fully automated means. Recently there has been a growing interest in machine translation projects [Nirenburg 87]. Many of these projects do not aim at full automation and people involved are pursuing ways of using human expertise in the translation process. Therefore they are willing to consider interactive approaches to disambiguation [Tomita 86].

It must be kept in mind that the translation process does not necessarily require full understanding of the text. Many ambiguities may be preserved during translation [Pericliev 84], and thus should not be presented to the user (human translator) for resolution. This approach not only saves unnecessary interaction but also yields a more true-to-the-source translation. Of course the preservability of an ambiguity on translation

is source-and-target language dependent. For example, the PP attachment ambiguity in the classic

(9) I saw a man in the park with a telescope.

is preserved when translating into Hebrew, since the attachment of the prepositional phrase is similar in both languages, and there is a Hebrew word equivalent to "**with**" that can be used in the senses of both "to **see with** an instrument" and "to **be with** an instrument". The same ambiguity cannot be preserved when translating into Finnish in which the preposition affects the verb it refers to.

Given that translation requires different disambiguation than general language understanding, and given that the special handling of ambiguities within the context of M(A)T is not only source language oriented but target dependent as well, it makes sense to look at the problem of disambiguation in the special context of M(A)T.

The natural place to resolve ambiguities is in the parser, where the power of syntax-directed techniques may be fully exploited, and where the early recognition of wrong parses improves the efficiency of the parser. A very elegant such solution is presented in [Tomita 86]. When dealing with machine translation projects, this approach may present several problems. In many cases translation projects use an already existing parser; thus changing or augmenting the parser may not be feasible. Such parsers are usually general purpose [Jensen 86], and as such are not concerned with translation-oriented disambiguation. This is especially valid for projects translating into different languages, where the same parser may be used [Peterson 80] but the need for different kinds of disambiguation may arise. Therefore it is worthwhile to study the approach of an independent translation-oriented module.

It should be noted that, as observed in [Tomita 86], interacting with the user after all trees have been found yields better organized questions. The parser we use filters out some of the trees automatically and presents all attachment ambiguities on one tree. Typical sentences yield 1-5 different parses with average of about 1.3 parses per sentence, thus the loss of efficiency is not consequential.

Although this paper deals with translation oriented disambiguation, the interaction with the user does not assume or require target language knowledge on behalf of the user (only on behalf of the system builder).

2. Background

The current work is motivated by the MENTOR project, under which several groups in European IBM Scientific Centers are collaborating on M(A)T research, focusing on English as a source language, and the different native languages as target languages. Since this paper originates from Israel, the examples refer to the English-to-Hebrew pair.

All groups participating in the MENTOR project use PEG [Jensen 86], which is a wide coverage syntactic parser, to process English input sentences. The output of the parser is one, or several trees, decorated by syntactic information gathered during parsing, e.g. tense, number, etc. and by some semantic markers provided by the English dictionary, e.g., Human, Concrete, etc.. However the bilingual transfer component expects as input **one** parse tree with as much syntactic and semantic information as the state of the art of

parsing provides. PEG suggests one preferred parse, but only as based on heuristic ranking [Heidorn 82].

This paper describes the concept of an interactive disambiguation module which is logically located between the parser and the transfer phase. The function of this module is to bridge the gap between several possible trees and/or other ambiguities, and one well-defined tree that may be satisfactorily translated. The gap is bridged by means of interaction with the user. True to the concepts presented above, only ambiguities that may not be preserved on translation to the specific target language are resolved.

3. Disambiguation by Rephrasing

The proposed technique for interactive disambiguation by rephrasing is based on three assumptions:

1. Ambiguous sentences are very frequently a consequence of a delicate balance between words that may have different parts of speech, and implicit hidden information that may be recovered in different ways. In both cases, a minor change in the sentences that disrupts that balance, or that makes the hidden information explicit, yields a non-ambiguous sentence preserving one of the possible meanings of the original ambiguous sentence.
2. It is easy for a human reader to decide whether or not a paraphrase of a sentence preserves the intended meaning of the original sentence. Furthermore, if the original sentence is not ambiguous to humans, the paraphrase on the wrong meaning yields in many cases a nonsense sentence which is even easier for a human to identify.
3. It is not necessary for a system to fully understand the source of an ambiguity in order to identify it and decide whether or not it may be preserved on translation. Full understanding is needed in order to select the correct interpretation, a task which in this case is left to the user.

Thus, a sentence is first parsed by a source-oriented general purpose parser, using a source language lexicon. Studying the several trees produced by the parser, special patterns stored in the disambiguation module, and syntactic information from the bilingual and target language dictionaries, potential ambiguities are identified. The preservability of the ambiguity on translation is tested. If the ambiguity is not found to be preservable, the sentence is paraphrased, and the paraphrases are presented to the user. According to the user's response, the appropriate input is forwarded to the next phase of translation.

The current work is restricted to four kinds of ambiguities (others are hard either to identify or to rephrase):

1. categorial ambiguity,
2. structural ambiguity,
3. word sense ambiguity and
4. implicit ambiguity of certain types.

The paraphrasing procedure consists of four parts:

1. identifying the ambiguity,
2. testing the preservability of the ambiguity,
3. rephrasing the sentence and

4. evaluating the answer and selecting the correct meaning.

For each kind of ambiguity, each step of the algorithm is slightly different. In the following paragraphs the various stages of the algorithm and their application to each kind
of ambiguity are discussed.

Since this work is done within the framework of a multi-language project, it is of interest
to observe which parts of the algorithm are target-language independent, or in other words common to all projects translating form the same **source** language, and which parts
are particular to a **pair** of languages. It is gratifying to find out that some parts are completely **general,** as summarized in the following table:

	Identifying	Testing	rephrasing	evaluating
Categorial	General	Pair	Source	General
Structural	General	Pair	Source	General
Word sense	Pair	——	Pair	General
Implicit	Pair	Pair	Source	Pair

3.1 Identifying Ambiguities

The first step in the rephrasing procedure is the identification of the ambiguities. The algorithms for identifying categorial and structural ambiguities are common to all ambiguities in each class, (i.e., the pair involved, e.g noun-verb or adjective-adverb, etc., is
not hard-coded into the algorithm). On the other hand, for identifying word-sense and implicit ambiguities, each case has to be known in advance and handled separately. Of course, like many tasks in natural language processing in general, and in machine translation in particular, writing a wide coverage system is very labor-intensive.

Identifying Categorial Ambiguities

A parallel scan of the leaves of two parse trees yields a list of all words which
are assigned different **poss.**

Identifying Structural Ambiguities

A parallel scan of two parse trees in a top-down order locates the first difference between the two trees.

Identifying Word Sense Ambiguities

The MENTOR project uses an active bilingual dictionary [Golan 88]. The selection of the target language word is done according to its syntactic environment in the source language parse tree. While writing the entry for a word, if two senses may not be differentiated by syntactic means, an ambiguity
is defined.

(10) He **held** a radical view.
(11) He **held** a radical conference.

In both cases **"held"** has a non-human, non-concrete direct object, but the Hebrew translation of the verb "hold" is different in each case. If the verb **"hold"** is encountered in this syntactic environment, an ambiguity is identified.

This word-by-word handling requires a considerable amount of work. In the MENTOR design, the active bilingual dictionary plays a major role, and each entry requires serious consideration in order to identify the syntactic environment which dictates the selection of a certain translation. Thus, actually listing the alternatives for which no automatic selection can be made is a relatively easy part of the dictionary creation.

Identifying Implicit Ambiguities

Patterns that are potentially ambiguous are defined. Example of several patterns are:

- VERB(BASE=BE) ADJ INFINITIVAL-CLAUSE (The chicken **are ready to** eat.)
- NP^2 NP* NOUN (Peterson presented his paper in the **desert language society symposium.)**
- ADJ NP^2 NP* NOUN (I take part in the **artificial intelligence project.)**

The tree is searched for such patterns. Collecting the list of patterns that are potentially ambiguous is not a straightforward task. It seems that the list grows as more sentence are incorrectly translated as a result of assuming the wrong meaning.

The parser determines to a large extent the cut between structural and implicit ambiguities. PEG for instance yields only one parse for a compound noun, thus we classify it as an implicit ambiguity, whereas a different parser could list several parses, which would then classify the same phenomena under structural ambiguity.

3.2 Preservability Test

When an ambiguity is identified, its preservability in the target language is tested. Note that the potential of preserving the ambiguity depends highly on the given language pair.

Preserving Categorial Ambiguities

Some **poss,** for example *"ing"* adjectives and verbs, that are interchangeable in English are interchangeable in Hebrew as well.

(12) The book is **interesting.**

Only conflicting **poss** are further dealt with.

Preserving Structural Ambiguities

Some structural ambiguities such as many cases of preposition attachment *(example* 9) may be preserved. Another example is the dual role of the word "**that**" denoting both a relative clause and a complementary clause.

(13) We signaled the guide that we could not hear.

Hebrew has an equivalent word that can be used in both senses.

Preserving Word Sense Ambiguities

Due to the technique used for identifying word sense ambiguity, as part of the entry in the bilingual dictionary,. these kinds of ambiguities may not be preserved.

Preserving Implicit Ambiguities

Many patterns of this sort, such as that in *example 1* may never be preserved. Those which may be preserved have their own algorithm for testing preservability. For example, an implicit pronoun reference *(example* 8) may be preserved if the relevant Hebrew nouns agree in number and gender.

3.3 Rephrasing the Sentence

Once an ambiguity is found to be unpresentable, a paraphrase is generated. When feasible, paraphrases are generated for both possible meanings, in order to accentuate the different meanings. In that case, the user is requested to choose one of the paraphrases. When only one paraphrase can be automatically generated, the user is required to state whether or not the original intention is preserved.

Rephrasing Categorial Ambiguities

Depending on the conflicting **poss** of a word or a series of words, one or two paraphrases are generated. *Example* 1 is identified by the following conflict:

$N_1\ V_2\ N_3\ N_4$ vs $N_1\ N_2\ V_3\ N_4$.

In this case the paraphrases are

the $N_1\ V_2$ *the* N_4 vs *the* $N_2\ V_3$ *the* N_4,

which yield:

(14) The management requests the information.
(15) The requests control the information.

Rephrasing Structural Ambiguities

According to the kind of conflict that occurs between two trees, one or two paraphrases are generated. In *example* 3 where the adverb may be attached either to the subordinate or to the main clause, the ambiguity could be avoided by correct punctuation. Thus, the two possibilities are presented to the user:

(16) Businessmen who are afraid to take risks , frequently lose out to their competitors.
(17) Businessmen who are afraid to take risks frequently , lose out to their competitors.

For *example* 4, the adjective is distributed to both NPs, to yield:

(18) Good boys and good girls go to heaven.

Rephrasing Word Sense Ambiguities

The entry in the bilingual dictionary has a synonym for each sense that cannot be disambiguated by its syntactic environment. The word in question is replaced by the alternative synonyms. For *example* 11, the paraphrases would be

(19) He convened a radical conference.
(20) He maintained a radical conference.

It should be emphasized that although the word sense disambiguation is triggered by the bilingual dictionary (written by experts in both source and target language), the user is presented by source language alternatives only.

Rephrasing Implicit Ambiguities

Each pattern for an implicit ambiguity has its own rephrasing algorithm. For the pattern VERB(base = be) ADJ INFINITIVAL-CLAUSE, in *example 7,* the infinitival clause is made passive.

(21) The chicken are ready to be eaten.

Note that for the same pattern with an intransitive verb, as in *example 22,* a nonsense sentence is generated, as shown in *example* 23.

(22) John is ready to sleep.
(23) John is ready to be slept.

3.4 Evaluating the Human's Answer

The human translator is presented with one or two paraphrases and has to state whether the meaning of the original sentence is preserved or which paraphrase preserves the original meaning. The answer then has to be assessed by the system.

Evaluating Categorial Ambiguities

All trees with conflicting assignment of **poss** are discarded.

Evaluating Structural Conflict

All trees with conflicting structures are discarded.

Evaluating Word Sense Selection

The correct target language translation is attached to the node in question.

Evaluating Implicit Conflicts

The explicit information is attached to the relevant nodes (e.g., the number and gender of a pronoun, the implicit preposition of compound nouns, etc.)

4. Conclusion

One approach for machine translation projects to overcome the problem of ambiguities, is to build in interaction with the user. Since many ambiguities may be preserved during translation, ambiguities that cannot be translated should be identified as such, and presented to the human translator in a way which will make the interactive

disambiguation task as convenient as possible. The technique of rephrasing a sentence to suit only one meaning was presented. The four main phases of the interactive disambiguation process, Identification, Preservability Test, Rephrasal, and Evaluation, were discussed in the context of four basic kinds of ambiguities, categorial, structural, word sense, and implicit.

We realize that the method presented may be inapplicable to some cases of ambiguity, and that even where it applies, more algorithms are needed for different cases. Nevertheless, we believe that the principles hereby advocated establish a firm base for the development of a software component that filters out many cases of wrong interpretations. Such a component could enhance the quality of the output generated by a machine translation system.

References

[Birnbaum 85] Birnbaum L., "Lexical Ambiguity as A Touchstone for Theories of Language Analysis," *IJCAI 85,* 1985, pp. 815-820.

[Dahlgren 86] Dahlgren K., and J. McDowell, "Using Commonsense Knowledge to Disambiguate Prepositional Phrase Modifiers," *AAAI-86,* Vol. 1, 1986, pp. 589-593.

[Duffy 86] Duffy G., "Categorial Disambiguation," *AAAI-86,* Vol. 2, 1986, pp. 1079-1082.

[Golan 88] Golan I., S. Lappin and M. Rimon, "An Active Bilingual Lexicon for Machine Translation," To appear in *Proc. of COLING'88, August 1988.*

[Heidorn 82] Heidorn E. G., "Experience with as Easily Computed Metric for Ranking Alternative Parsers," *ACL Meeting,* June 1982.

[Hirst 87] *Semantic Interpretation and the resolution of ambiguity ,* G. Hirst, Cambridge University Press, 1987.

[Jensen 86] Jensen K., "PEG 1986: A Broad-Coverage Computational System for English," Technical Report, *IBM T.J. Watson Research Center,* 1986.

[Milne 86] Milne R., "Resolving Lexical Ambiguity in a Deterministic Parser," *Computational Linguistics,* Vol. 12, No. 1, January-March 1986, pp. 1-12.

[Nirenburg 87] *Machine Translation: Theoretical and Methodological Issues,* Sergei Nirenburg (ed.), Cambridge University Press, 1987.

[Pericliev 84] Pericliev V., "Handling Syntactical Ambiguity in Machine Translation," *COLING'84,* July 1984, pp. 521-524.

[Peterson 80] Peterson J.E., "Word Sense Selection in A One-To-Many, Interactive Computer-Assisted Translation System," *6th Annual Symposium of the Desert Language and Linguistic Society,* Utah, 1980, pp. 150-163.

[Tomita 86] Tomita M., "Sentence Disambiguation by Asking," *Computer and Translation,* Vol. 1, 1986, pp. 39-51.