[MT Summit III, Washington, DC, USA, July 1991]

ULTRA: A Multilingual Machine Translator

David Farwell and Yorick Wilks
Computing Research Laboratory
New Mexico State University
Box 30001, Las Cruces, NM 88003

Abstract

ULTRA (Universal Language TRAnslator) is a multilingual, interlingual machine translation system currently under development at the Computing Research Laboratory at New Mexico State University. It translates between five languages (Chinese, English, German, Japanese, Spanish) with vocabularies in each language based on approximately 10,000 word senses. The major design criteria are that the system be robust and general purpose with simple to use utilities for customization to suit the needs of particular users.

This paper describes the central characteristics of the system: the intermediate representation, the language components, semantic and pragmatic processes, and supporting lexical entry tools.

1 Introduction

ULTRA (Universal Language TRAnslator) is a multilingual, interlingual machine translation system under development at the Computing Research Laboratory at New Mexico State University. It currently translates between five languages (Chinese, English, German, Japanese, Spanish) with vocabularies in each language based on about 10,000 word senses. The major design criteria are that the system be robust and general purpose with simple to use utilities for customization to suit the needs of particular users.

Its special features include:

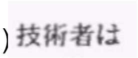
- a multilingual system with a language-independent (interlingual) system for representing expressions as elements of linguistic acts;
- bidirectional, Prolog grammars for each language incorporating semantic and pragmatic constraints;
- use of relaxation techniques to provide robustness by giving preferred or "near miss" translations;
- language-independent semantic and pragmatic procedures for disambiguation and interpreting elipted information;
- access to large machine dictionaries for rapid up-scaling of size and coverage.

2 The system of intermediate representation

The interlingual representation (IR) has been designed to reflect our assumption that what is universal about language is that it is used to perform acts of communication: asking questions, describing the world, expressing one's thoughts, getting people to do things, warning them not to do things, promising that things will get done and so on. Translation, then, can be viewed as the use of the target language to perform the same act as that which was performed using the source language. The IR serves as the basis for analyzing or for generating expressions as elements of such acts in each of the languages in the translation system.

The representation has been formulated on the basis of an on-going cross-linguistic comparative analysis of hand-generated translations with respect to the kinds of information necessary for selecting the appropriate forms of equivalent expressions in the different languages in the system. We have looked at a number of different types of communication including expository texts, business letters, e-mail messages and dialogues. This, coupled with the fact that the languages selected for the initial development stage are of different historical and typological background, has led to a solid foundation for developing a flexible and complete descriptive framework.

By way of example, the Japanese expression in (1),

(1) 
技術者は

```
  engineer-top
```

is associated with the intermediate representation in (2).

```
(2) [arg, [g_rel,subj],
          [k_rel,agnt],
          [t_rel,top],
     [ent,   [type,nrm],
             [class, human],
             [agree,A] ,
             [det,spin],
             [quant,unq],
             [e_desc,engineer1_1]]]
```

Without going into the specifics of the representational framework, (2) is intended to express the fact that the expression is fills the role of subject argument, **arg ... subj**, in some larger proposition. In addition, it has the semantic role of agent, **arg ... agnt**, with respect to some predicate and the functional role of topic, **arg ... top**, within some utterance. The intended sense of the expression has the properties describing a general class of objects, **ent ... nrm**, which are human, **ent ... human**. In addition, the objects referred to on this occasion are specified by a larger context with respect to number, **ent ... A**, assumed by the speaker to be identifiable by the addressee, **ent ... spin**, and are unique within some larger context, **ent ... unq**. The sense, itself, is represented by the entity descriptor, **e_desc, engineerl_l**.

As noted above, an IR is a representation of the explicit information in the context of the expression being processed. This includes the referential, stylistic and communicative aspects of an utterance in a language in so far as these are reflected by the form of the expression uttered. It does not include information inferable from such explicit information. While IRs are "language neutral", they are representations of linguistic acts rather than representations of the conceptual contents of expressions.

3 The language components

Each individual language system is independent of all other language systems within ULTRA and has its own procedures for associating the expressions of the language with the appropriate IRs. These independent systems communicate by handing each other IRs so no transfer takes place. Independence of the language particular systems is of both theoretical and practical interest. In order to be successful, the intermediate representations must be "correct" in the sense that they contain all the relevant information for choosing an equivalent form in any of the different languages in the system.

Of practical interest, any new language may be added to the translation system at any time without having unpredictable or negative side effects on the language systems previously developed or on the overall performance of the system. This not only allows the different language system developers freedom in choosing the class of grammar and the type of parser and generator they feel most comfortable with, but it also lends itself to the rapid extension to new translation applications and great flexibility in the design of supporting software.

Thus far, all of the systems fall within the class of unification grammars. Prototype systems have been based on a number of different grammatical approaches including Semantic Definite Clause Grammar [Pereira & Warren, 1980; Huang, 1988], Case Grammar [Nagao, et al. 1985], Categorial Grammar [Uszkoreit, 1986], as well as a semantic constituent structure grammar under development at the CRL. All of these utilize both semantic and syntactic constraints for dealing with ambiguity.

The Spanish component, for instance, currently takes the form of a Semantic Definite Clause Grammar which is essentially a context-free phrase structure grammar with complex categories. Viewed procedurally, it is a top-down, depth-first, left-to-right, unification-based parser/generator of a subset of Spanish expressions.

A typical rule schema takes the following form:

```
(3)    category_0  (Fl,  F2, ....
                   [cat_0,
                      Substruc_l,
                      Substruc_2],
                    String_in,
                    Rest)  : -
       category_1   (Fl,   ....
                    Substruc_l,
                    String_in,
                    String_in_rest),
       category_2  (F2,   ....
                    Substruc_2,
                    String_in_rest,
                    Rest).
```

This rule stipulates that there is a correspondence for Spanish, sanctioned by the rule for **category_0**, between structures of the form **[cat_0, Substruc_l, Substruc_2]** and expressions of the form **String_in** if and only if there is a correspondence, sanctioned by a rule for **category_1**, between structures of the form **Substruc_l** and expressions of the form of some initial substring of **String-in** and there is also a correspondence, sanctioned by the rule **category_2**, between structures of the form **Substruc_2** and expressions of the form **String_in_rest** which spans the remaining substring of String_in, such that all features, **Fl**, **F2**, and so on are consistent.

Through with the use of logic programming techniques, we have been successful in developing bidirectional parser/generators. The same algorithm either accepts an expression as input and provides an IR as output or accepts an IR as input and provides an expression as output, Bidirectionality, or symmetry, is introduced through the explicit mention of the structural schema together with the implicit extra two arguments of the predicate, the string under analysis and the string which is left over after the analysis, in every rule of the grammar. This essentially converts each rule into an equivalence statement between structures and expressions.

The point may be made more clearly, perhaps, through an example. In reference to the schematic rule in (3), suppose it is taken as a rule which equates IRs for specified entity descriptions with expressions in Spanish. Specifically, suppose the rule is called during analysis with the variables instantiated as in (4),

```
(4)    category_0(Agree,   Gender,
                 [cat_0,
                   Substruc_1,
                   Substruc_2],
                 [el,avion], R)  : -
        category_1 (Agree, Gender,
                   Substruc_1,
                    [el,avion],
                   String_in_rest),
        category_2 (Agree, Gender,
                   Substruc_2,
                   String_in_rest, R).
```

The first subgoal is to show that some initial part of the string *el avion* can be equated with an IR under **category_l.** Now suppose the rule succeeds with the variables bound as in (5):

```
(5)  category_l  (ts,  m,
                 [e_spec,  generic],
                 [el, avion],
                 [avion]).
```

The second subgoal is processed similarly and, when it succeeds, results in **category_0** succeeding with its variables bound as follows.

```
(6)  category_0  (ts, m,
                 [cat_0,
                   [e_spec, generic],
                   [e_desc, aircraft0_0]],
                 [el,avion], []).
```

Now suppose that, during generation, (3) is called with the variables instantiated as in (7).

```
(7)  category_0  (Agree,  Gender,
                 [cat_0,
                    [e_spec, generic],
                    [e_desc, aircraft0_0]],
                  String_in, []) : -
        category_l (Agree, Gender,
                   [e_spec,generic],
                   String_in,
                   String_in_rest),
        category_2 (Agree, Gender,
                   [e_desc, aircraft0_0],
                   String_in_rest, []).
```

The first subgoal is to show that **[e_spec, generic]** can be equated with some string under **category_l**. Let that subgoal succeeds with variables bound as in (8):

```
(8)  category_l  (ts, m,
                 [e_spec,  generic] ,
                 [el   |  String_in_rest],
                 String_in_rest).
```

Again, the second subgoal is processed similarly, and, assuming it succeeds, results in category_0 succeeding with its variables bound as follows.

```
(9)  category_0 (ts, m,
                [cat_0,
                  [e_spec,  generic],
                  [e_desc, aircraft0_0]],
                [el, avion], [] ) .
```

Since all the rules in the system are of this general formal, symmetry, or bidirectionality is maintained at every level.

Lexical entries in the system have two parts: a language particular entry corresponding the graphic form used to express some sense, and an intermediate representational element corresponding loosely to a word sense token for the sense expressed. All the entries take the form of simple Prolog unit clauses of the general form in (10):

```
(10)  category  (Form,  Fl,  F2,   ...).
```

where **Fl, F2**, and so on are constraints. For language particular entries, these are generally syntactic constraints associated with the graphic form, Form, such as the gender of a noun, whether a verb is reflexive, and so on. In addition, the final argument represents the IR sense the form is used to express and acts as a pointer to the IR lexicon as well as any lexical resources which might be available. For IR entries, the features correspond to universal semantic and pragmatic constraints associated with the sense such as the classification of an entity as countable or non-countable, the semantic case structure of a relation, and so on.

4 Relaxation

Although the CRL systems are capable of handling a wide range of phenomena, there will always be classes of “non-standard” input which will fall outside the system's normal capabilities. To deal with such input, we are developing a range of techniques falling under the rubric of “relaxation”. There are three basic cases for which techniques have been developed and are being implemented: grammatical relaxation, semantic relaxation, and structural relaxation.

For semantic relaxation we have introduced a special predicate which will identify a pairs pairs of semantic classes as comparable so that if there is no exact match between the semantic preference of some element and the semantic class of its dependent, a looser notion of semantic compatibility is expressed by the special predicate. For grammatical relaxation, the approach is to systematically remove specific grammatical constraints and reprocess the string. For structural relaxation, the string is reanalyzed as an arbitrary sequence of lower level constituent, If all of the above methods fail, a word-by-word translation is provided in order to present at least grammatical and semantic information of each lexical item in the input,

5 Independent semantic and pragmatic procedures

Because, for any given expression in some language-there may be several possible representations, we have developed procedures for choosing, given a context, the best IR from the set of possible IR's, removing the unintelligible IR's, and ordering the remaining IR's with respect to which is "preferred" in that context.

For instance, consider the problems of lexical disambiguation, a standard issue addressed in natural language systems that are based on Preference Semantics [Wilks, 1975, 1978] and Collative Semantics [Fass, 1988]. The English sentence in (11),

(11) *The speaker reached the central point of his paper.*

contains five ambiguous lexical items, *speaker, reach, central, point* and *paper* which must be resolved in order to translate the sentence into, say, German, The task of selecting a coherent combination of the possible senses is semantic in that it is based on what we know about the objects, properties and actions referred to and how they normally relate to one another in our predictable world. Consider the two possible interpretations of *reach* as “extend to” and as "achieve". Generally, we would assume that loudspeakers extend to places while orators and, less obviously, language users achieve things, namely, expressing arguments, issues, and so on. Thus, we are immediately beginning to build two different coherent scenarios. This general process is repeated with each new input "sense" until the reading with the greatest overall coherence is selected. In this case, it is likely that (11) is to be translated as in (12),

(12) *Der Redner hat die Hauptsache seines Vortrags erreicht.*

The procedures described are based on the model of interpretation embodied in the PREMO Preference Semantics parser [Slator, 1988] and Meta5 Collative Semantics analysis program [Helmreich, et al. 1990]. We are currently adapting them to operate on IRs.

6 Support tools

As regards lexical entries, the ULTRA system currently provides for either interactive or limited automatic entry. During interactive entry, the user is prompted for a minimum amount of information based on the linguistic context of the item being specified and provided with a number of on-line resources, including the *Longman Dictionary of Contemporary English* (LDOCE) [Proctor, et al. 1978], to aid in responding. Automatic entry is currently limited to the entry of new IR tokens.

For interactive entry, the user is guided through an active-forms based entry system or a menu-entry system. Implemented in LISP within the Gemacs environment, the systems allows for the statement of default specifications and co-occurrence restrictions between any field on any form in the system and any other field on any other form. Having completed the specification, the user is then presented with a full specification of the item for confirmation or, possibly, correction.

The items to be entered may be identified automatically, namely, through a preprocessing search for unknown spelling forms. If an unknown form is found, the item is marked and translation proceeds as usual. When the analysis system reaches the point where processing begins on the marked item, processing is interrupted and the user is prompted for a specification of the new item. This allows for the use of information from the context of the item thus reducing the amount of information that the user needs to provide. With the specification of the source language and IR item completed, a dummy target language item is constructed with the source language spelling form temporarily standing as the target language form. Processing continues and the translation is completed with the source language item appearing in the target language text. The user is again consulted, this time as to the appropriate target language item,

As for automatic entry, the Computing Research Laboratory is drawing upon extensive research in deriving semantic structures automatically from large machine-readable dictionaries [Slator, 1988; Wilks & Slator, 1989]. Much of the core IR lexicon has been derived from the 72,000 word senses in LDOCE. Codings form the machine readable version of the dictionary for such properties as semantic category, selection restrictions and so on have been used, either directly or indirectly, to generate partial specifications of some 7,000 IR tokens for the system. We are looking at the application of machine readable versions of bilingual dictionaries to the automatic entry of the individual language lexicons. These two entry techniques allows for the rapid upscaling of the size and coverage of the vocabularies and for tailoring them to the individual needs of the user.

7 Summary

Currently the prototype system produces word, phrase or sentence level translations and handles most basic declarative, interrogative and imperative structures, including conjoined and subjoined constructions, while dealing with various types of sense disambiguation and structurally dependent anaphora and ellipsis. Each language component was developed independently and reflects the individual preferences of the particular researcher toward the tasks of parsing and generating expressions in the language they were concerned with. All of the language components have vocabularies based on some 10,000 word senses. Input and output for the Chinese and Japanese systems may take the form of latin alphabet or characters. Input and output for the Spanish and German systems may contain the special characters associated with their alphabets or be restricted to a normal keyboard alphabet. Support facilities are being developed which permit the user to diagnose and debug lexical failure, diagnose and debug structural failures, and test the system over selected corpora. All the language components are implemented in Quintus Prolog running on SUN 3's and SUN 4's in both batch and interactive modes.

8 References

[Fass, 1988] Fass, D. Collative Semantics: A Semantics for Natural Language Processing. *Memoranda in Computer and Cognitive Science,* MCCS-88-118, Computing Research Laboratory, New Mexico State University, Las Cruces, NM.

[Helmreich, et al. 1990] Helmreich, S., Iverson, E., and Laroche, F. Modular Meta5: Further Research in Collative Semantics. *Memoranda in Computer and Cognitive* Science, MCCS-90-192. Computing Research Laboratory, New Mexico State University, Las Cruces, NM.

[Huang, 1990] Huang, X-M. Semantic analysis in XTRA, an English-Chinese machine translation system. *Computers and Translation,* 3, pp. 101-120.

[Nagao, et al. 1985] Nagao, M., Tsujii, J-C., and Nakamura, J-C. The Japanese government project for machine translation. *Computational Linguistics,* 11:2-3, pp. 91-110.

[Pereira & Warren, 1980] Pereira, F. and Warren, D. Definite clause grammars for language analysis-a survey of the formalism and a comparison with augmented transition networks. *Artificial Intelligence,* 13, pp. 231-278.

[Procter et al, 1978] P. Procter, et al. *Longman Dictionary of Contemporary English.* Harlow, UK: Longman Group Limited.

[Slator, 1988] Slator, B. Lexical Semantics and Preference Semantics Analysis. *Memoranda in Computer and Cognitive Science,* MCCS-88-143, Computing Research Laboratory, New Mexico State University, Las Cruces, NM.

[Uszkoreit, 1986] Uszkoreit, H. Categorial Unification Grammars. Report 66, Center for the Study of Language and Information, Stanford, CA.

[Wilks, 1975] Wilks, Y. A preferential Pattern-Seeking Semantics for Natural Language inference. *Artificial Intelligence,* 6, pp. 53-74.

[Wilks, 1978] Wilks, Y. Making Preferences More Active. *Artificial Intelligence,* 10, pp. 1-11.

[Wilks & Slator, 1989] Wilks, Y. and Slator, B. PREMO: parsing by conspicuous lexical consumption. *Proceedings of the International Workshop on Parsing Technologies,* M. Tomita (ed.), Carnegie Mellon University. To be published by Morgan Kaufman.