Multi-lingual Translation of Spontaneously Spoken Language in a Limited Domain

Alon Lavie, Donna Gates, Marsal Gavaldà,
Laura Mayfield, Alex Waibel and Lori Levin
Center for Machine Translation
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213
email : lavie@cs.cmu.edu

Abstract

JANUS is a multi-lingual speech-to-speech translation system designed to facilitate communication between two parties engaged in a spontaneous conversation in a limited domain. In an attempt to achieve both robustness and translation accuracy we use two different translation components: the GLR module, designed to be more accurate, and the Phoenix module, designed to be more robust. We analyze the strengths and weaknesses of each of the approaches and describe our work on combining them. Another recent focus has been on developing a detailed end-to-end evaluation procedure to measure the performance and effectiveness of the system. We present our most recent Spanish-to-English performance evaluation results.

1 Introduction

JANUS is a multi-lingual speech-to-speech translation system designed to facilitate communication between two parties engaged in a spontaneous conversation in a limited domain. In this paper we describe the current design and performance of the machine translation module of our system. The analysis of spontaneous speech requires dealing with problems such as speech disfluencies, looser notions of grammaticality and the lack of clearly marked sentence boundaries. These problems are further exacerbated by errors of the speech recognizer. We describe how our machine translation system is designed to effectively handle these and other problems. In an attempt to achieve both robustness and translation accuracy we use two different translation components: the GLR module, designed to be more accurate, and the Phoenix module, designed to be more robust. Both modules follow an interlingua-based approach. The translation modules are designed to be language-independent in the sense that they each consist of a general processor that applies independently specified knowledge about different languages. This facilitates the easy adaptation of the system to new languages and domains. We analyze the strengths and weaknesses of each of the translation approaches and describe our work on combining them. Our current system is designed to translate spontaneous dialogues in the Scheduling domain, with English, Spanish and German as both source and target languages. A recent focus has been on developing a detailed end-to-end evaluation procedure to measure the performance and effectiveness of the system. We describe this procedure in the latter part of the paper, and present our most recent Spanish-to-English performance evaluation results.

2 System Overview

The JANUS System is a large scale multi-lingual speech-to-speech translation system designed to facilitate communication between two parties engaged in a spontaneous conversation in a limited domain. A diagram of the architecture of the system is shown in Figure 1. The system is composed of three main components: a speech recognizer, a machine translation (MT) module and a speech synthesis module. The speech recognition component of the system is described elsewhere (Woszczyna et al. 1994). For speech synthesis, we use a commercially available speech synthesizer.

The MT module is composed of two separate translation sub-modules which operate independently. The first is the GLR module, designed to be more accurate. The second is the Phoenix module, designed to be more robust. Both modules follow an interlingua-based approach. The source language input string is first analyzed by a parser, which produces a language-independent interlingua content representation. The interlingua is then passed to a generation component, which produces an output string in the target language.

The discourse processor is a component of the GLR translation module. It disambiguates the speech act of each sentence, normalizes temporal

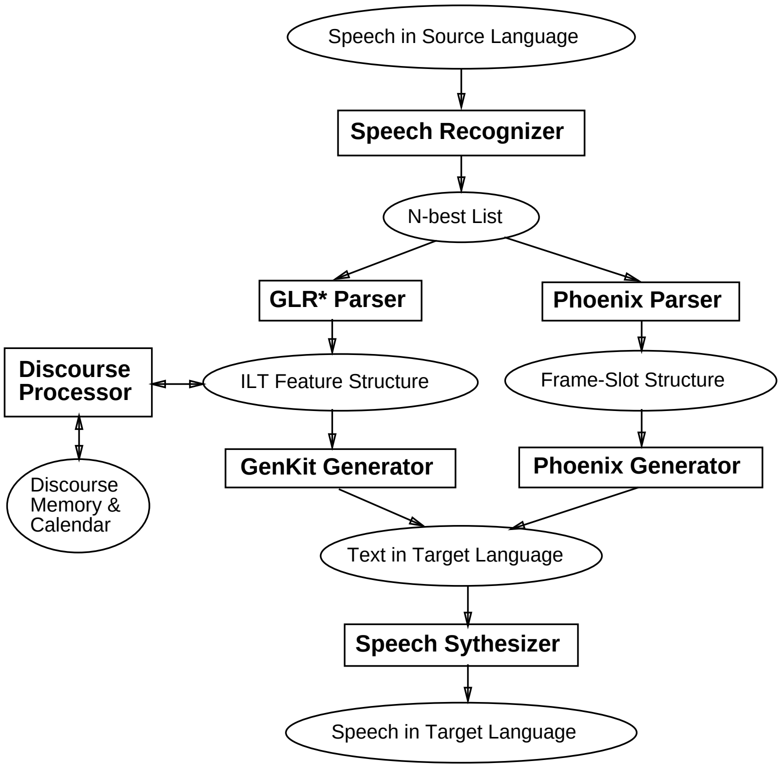

Figure 1: **The JANUS System**

expressions, and incorporates the sentence into a discourse plan tree. The discourse processor also updates a calendar which keeps track of what the speakers have said about their schedules. The discourse processor is described in greater detail elsewhere (Rosé et al. 1995).

3 The GLR Translation Module

The GLR* parser (Lavie and Tomita 1993; Lavie 1994) is a parsing system based on Tomita's Generalized LR parsing algorithm (Tomita 1987). The parser skips parts of the utterance that it cannot incorporate into a well-formed sentence structure. Thus it is well-suited to domains in which nongrammaticality is common. The parser conducts a search for the maximal subset of the original input that is covered by the grammar. This is done using a beam search heuristic that limits the combinations of skipped words considered by the parser, and ensures that it operates within feasible time and space bounds.

The GLR* parser was implemented as an extension to the GLR parsing system, a unification-based practical natural language system (Tomita 1990). The grammars we develop for the JANUS system are designed to produce feature structures that correspond to a frame-based language-independent representation of the meaning of the input utterance. For a given input utterance, the parser produces a set of interlingua texts, or ILTs. The main components of an ILT are the speech act (e.g., `suggest`, `accept`, `reject`), the sentence type (e.g., `state`, `query-if`, `fragment`), and the main semantic frame (e.g., `free`, `busy`). An example of an ILT is shown in Figure 2. A detailed ILT Specification was designed as a formal description of the allowable ILTs. All parser output must conform to this ILT Specification. The GLR unification based formalism allows the grammars to construct precise and very detailed ILTs. This in turn allows the GLR translation module to produce highly accurate translations for well-formed input.

The GLR* parser also includes several tools designed to address the difficulties of parsing spontaneous speech. To cope with high levels of ambiguity, the parser includes a statistical disambiguation module, in which probabilities are attached directly to the actions in the LR parsing table. The parser can identify sentence boundaries within each hypothesis with the help of a statistical method that determines the probability of a boundary at each point in the utterance. The parser must also determine the "best" parse from among the different parsable subsets of an input. This is done using a collection of parse evaluation measures which are combined into an integrated heuristic for evaluating and ranking the parses produced by the parser. Additionally, a parse quality heuristic allows the parser to self-

```
((frame *free)
 (who ((frame *i)))
 (when ((frame *simple-time)
        (day-of-week wednesday)
        (time-of-day morning)))
 (a-speech-act (*multiple* *suggest *accept))
 (sentence-type *state)))

Sentence: I could do it Wednesday morning too.
```

Figure 2: An Example ILT

judge the quality of the parse chosen as best, and to detect cases in which important information is likely to have been skipped.

Target language generation in the GLR module is done using GenKit (Tomita and Nyberg 1988), a unification-based generation system. With well-developed generation grammars, GenKit results in very accurate translation for well-specified ILTs.

4 The Phoenix Translation Module

The JANUS Phoenix translation module (Mayfield et al. 1995) is an extension of the Phoenix Spoken Language System (Ward 1991; Ward 1994). The translation component consists of a parsing module and a generation module. Translation between any of the four source languages (English, German, Spanish, Korean) and five target languages (English, German, Spanish, Korean, Japanese) is possible, although we currently focus only on a few of these language pairs.

Unlike the GLR method which attempts to construct a detailed ILT for a given input utterance, the Phoenix approach attempts to only identify the key semantic concepts represented in the utterance and their underlying structure. Whereas GLR* is general enough to support both semantic and syntactic grammars (or some combination of both types), the Phoenix approach was specifically designed for semantic grammars. Grammatical constraints are introduced at the phrase level (as opposed to the sentence level) and regulate semantic categories. This allows the ungrammaticalities that often occur between phrases to be ignored and reflects the fact that syntactically incorrect spontaneous speech is often semantically well-formed.

The parsing grammar specifies patterns which represent concepts in the domain. The patterns are composed of words of the input string as well as other tokens for constituent concepts. Elements (words or tokens) in a pattern may be specified as optional or repeating (as in a Kleene star mechanism). Each concept, irrespective of its level in the hierarchy, is represented by a separate grammar file. These grammars are compiled into Recursive Transition Networks (RTNs).

The interlingua meaning representation of an input utterance is derived directly from the parse tree constructed by the parser, by extracting the represented structure of concepts. This representation is usually less detailed than the corresponding GLR ILT representation, and thus often results in a somewhat less accurate translation. The set of semantic concept tokens for the Scheduling domain was initially developed from a set of 45 example English dialogues. Top-level tokens, also called slots, represent speech acts, such as suggestion or agreement. Intermediate-level tokens distinguish between points and intervals in time, for example; lower-level tokens capture the specifics of the utterance, such as days of the week, and represent the only words that are translated directly via lookup tables.

The parser matches as much of the input utterance as it can to the patterns specified by the RTNs. Out-of-lexicon words are ignored, unless they occur in specific locations where open concepts are permitted. A word that is already known to the system, however, can cause a concept pattern not to match if it occurs in a position unspecified in the grammar. A failed concept does not cause the entire parse to fail. The parser can ignore any number of words in between top-level concepts, handling out-of-domain or otherwise unexpected input. The parser has no restrictions on the order in which slots can occur. This can cause added ambiguity in the segmentation of the utterance into concepts. The parser uses a disambiguation algorithm that attempts to cover the largest number of words using the smallest number of concepts.

Figure 3 shows an example of a speaker utterance and the parse that was produced using the Phoenix parser. The parsed speech recognizer output is shown with unknown (-) and unexpected (*) words marked. These segments of the input were ignored by the parser. The relevant concepts, however, are extracted, and strung together they provide a general meaning representation of what the speaker actually said.

Generation in the Phoenix module is accomplished using a simple strategy that sequentially generates target language text for each of the top level concepts in the parse analysis. Each concept has one or more fixed phrasings in the target language. Variables such as times and dates are extracted from the parse analysis and translated directly. The result is a meaningful translation, but can have a telegraphic feel.

5 Combining the GLR and Phoenix Translation Modules

5.1 Strengths and Weaknesses of the Approaches

As already described, both the GLR* parser and the Phoenix parser were specifically designed to handle the problems associated with analyzing spontaneous speech. However, each of the ap-

Original utterance:
SÍ QUÉ TE PARECE TENGO EL MARTES DIECIOCHO Y EL MIÉRCOLES DIECINUEVE LIBRES TODO EL DÍA PODRÍAMOS IR DE MATINÉ O SEA EN LA TARDE VER EL LA PELÍCULA
(Roughly *"Yes what do you think I have Tuesday the eighteenth and Wednesday the nineteenth free all day we could go see the matinée so in the afternoon see the the movie."*)

As decoded by the recognizer:
%NOISE% SI1 QUE1 TE PARECE %NOISE% TENGO EL MARTES DIECIOCHO Y EL MIE1RCOLES DIECINUEVE LIBRES TODO EL DI1A PODRI1AMOS IR DE MATINE1 %NOISE% O SEA LA TARDE A VER LA

Parsed:
%<S> si1 que1 te parece tengo el martes dieciocho y el mie1rcoles diecinueve libres todo el di1a podri1amos *IR *DE -MATINE1 o sea la tarde a ver LA %</S>

Parse Tree (≡ Semantic Representation):

[respond] ([yes] (SI1))

[your˙turn] (QUE1 TE PARECE)

[give˙info] ([my˙availability] (TENGO [temp˙loc] ([temporal] ([point] ([date] (EL [d˙o˙w] (MARTES)) [date] ([day˙ord] (DIECIOCHO) [conj] (Y) EL [d˙o˙w] (MIE1RCOLES)) [date] ([day˙ord] (DIECINUEVE))))) LIBRES))

[give˙info] ([my˙availability] ([temp˙loc] ([temporal] ([range] ([entire] (TODO)EL [unit] ([t˙unit] (DI1A)))))PODRI1AMOS))

[suggest] ([suggest˙meeting] ([temp˙loc] ([temporal] (O SEA [point] (LA [t˙o˙d] (TARDE))))A VER))

Generated:
English = <Yes what do you think? I could meet Tuesday eighteenth and Wednesday the nineteenth I could meet the whole day do you want to try to get together in the afternoon>

Figure 3: A Phoenix Spanish to English Translation Example

proaches has some clear strengths and weaknesses.

Although designed to cope with speech disfluencies, GLR* can gracefully tolerate only moderate levels of deviation from the grammar. When the input is only slightly ungrammatical, and contains relatively minor disfluencies, GLR* produces precise and detailed ILTs that result in high quality translations. The GLR* parser has difficulties in parsing long utterances that are highly disfluent, or that significantly deviate from the grammar. In many such cases, GLR* succeeds to parse only a small fragment of the entire utterance, and important input segments end up being skipped. [1] Phoenix is significantly better suited to analyzing such utterances. Because Phoenix is capable of skipping over input segments that do not correspond to any top level semantic concept, it can far better recover from out-of-domain segments in the input, and "restart" itself on an in-domain segment that follows. However, this sometimes results in the parser picking up and mis-translating a small parsable phrase within an out-of-domain segment. To handle this problem, we are attempting to develop methods for automatically detecting out-of-domain segments in an utterance (see section 7).

Because the Phoenix approach ignores small function words in the input, its translation results are by design bound to be less accurate. However, the ability to ignore function words is of great benefit when working with speech recognition output, in which such words are often mistaken. By keying on high-confidence words Phoenix takes advantage of the strengths of the speech decoder. At the current time, Phoenix uses only very simple disambiguation heuristics, does not employ any discourse knowledge, and does not have a mechanism similar to the parse quality heuristic of GLR*, which allows the parser to self-assess the quality of the produced result.

5.2 Combining the Two Approaches

Because each of the two translation methods appears to perform better on different types of utterances, they may hopefully be combined in a way that takes advantage of the strengths of each of them. One strategy that we have investigated is to use the Phoenix module as a back-up to the GLR module. The parse result of GLR* is translated whenever it is judged by the parse quality heuristic to be "Good". Whenever the parse result from GLR* is judged as "Bad", the translation is generated from the corresponding output of the Phoenix parser. Results of using this combination scheme are presented in the next section. We are in the process of investigating some more sophisticated methods for combining the two translation approaches.

6 Evaluation

6.1 The Evaluation Procedure

In order to assess the overall effectiveness of the two translation components, we developed a detailed end-to-end evaluation procedure (Gates et al. 1996). We evaluate the translation modules on both transcribed and speech recognized input. The evaluation of transcribed input allows us to assess how well our translation modules would function with "perfect" speech recognition. Testing is performed on a set of "unseen" dialogues, that were not used for developing the translation modules or training the speech recognizer.

The translation of an utterance is manually evaluated by assigning it a grade or a set of grades based on the number of sentences in the utterance. The utterances are broken down into sentences for evaluation in order to give more weight to longer utterances, and so that utterances containing both in and out-of-domain sentences can be judged more accurately.

Each sentence is classified first as either relevant to the scheduling domain (in-domain) or not rel-

[1] Recent work on a method for pre-breaking the utterance at sentence boundaries prior to parsing have significantly reduced this problem.

evant to the scheduling domain (out-of-domain). Each sentence is then assigned one of four grades for translation quality: (1) Perfect - a fluent translation with all information conveyed; (2) OK - all important information translated correctly but some unimportant details missing, or the translation is awkward; (3) Bad - unacceptable translation; (4) Recognition Error - unacceptable translation due to a speech recognition error. These grades are used for both in-domain and out-of-domain sentences. However, if an out-of-domain sentence is automatically detected as such by the parser and is not translated at all, it is given an "OK" grade. The evaluations are performed by one or more independent graders. When more than one grader is used, the results are averaged together.

6.2 Results

Figure 4 shows the evaluation results for 16 unseen Spanish dialogues containing 349 utterances translated into English. Acceptable is the sum of "Perfect" and "OK" sentences. For speech recognized input, we used the first-best hypotheses of the speech recognizer.

Two trends have been observed from this evaluation as well as other evaluations that we have conducted. First, The GLR translation module performs better than the Phoenix module on transcribed input and produces a higher percentage of "Perfect" translations, thus confirming the GLR approach is more accurate. This also indicates that GLR performance should improve with better speech recognition and improved pre-parsing utterance segmentation. Second, the Phoenix module performs better than GLR on the first-best hypotheses from the speech recognizer, a result of the Phoenix approach being more robust.

These results indicate that combining the two approaches has the potential to improve the translation performance. Figure 5 shows the results of combining the two translation methods using the simple method described in the previous section. The GLR* parse quality judgement is used to determine whether to output the GLR translation or the Phoenix translation. The results were evaluated only for in-domain sentences, since out-of-domain sentences are unlikely to benefit from this strategy. The combination of the two translation approaches resulted in a slight increase in the percentage of acceptable translations on transcribed input (compared to both approaches separately). On speech recognized input, although the overall percentage of acceptable translations does not improve, the percentage of "Perfect" translations was higher. [2]

[2] In a more recent evaluation, this combination method resulted in a 9.5% improvement in acceptable translations of speech recognized in-domain sentences. Although some variation between test sets is to be expected, this result strengthens our belief in the potential of this approach.

7 Conclusions and Future Work

In this paper we described the design of the two translation modules used in the JANUS system, outlined their strengths and weaknesses and described our efforts to combine the two approaches. A newly developed end-to-end evaluation procedure allows us to assess the overall performance of the system using each of the translations methods separately or both combined.

Our evaluations have confirmed that the GLR approach provides more accurate translations, while the Phoenix approach is more robust. Combining the two approaches using the parse quality judgement of the GLR* parser results in improved performance. We are currently investigating other methods for combining the two translation approaches. Since GLR* performs much better when long utterances are broken into sentences or sub-utterances which are parsed separately, we are looking into the possibility of using Phoenix to detect such boundaries. We are also developing a parse quality heuristic for the Phoenix parser using statistical and other methods.

Another active research topic is the automatic detection of out-of-domain segments and utterances. Our experience has indicated that a large proportion of bad translations arise from the translation of small parsable fragments within out-of-domain phrases. Several approaches are under consideration. For the Phoenix parser, we have implemented a simple method that looks for small islands of parsed words among non-parsed words and rejects them. On a recent test set, we achieved a 33% detection rate of out-of-domain parses with no false alarms. Another approach we are pursuing is to use word salience measures to identify and reject out-of-domain segments.

We are also working on tightening the coupling of the speech recognition and translation modules of our system. We are developing lattice parsing versions of both the GLR* and Phoenix parsers, so that multiple speech hypotheses can be efficiently analyzed in parallel, in search of an interpretation that is most likely to be correct.

Acknowledgements

The work reported in this paper was funded in part by grants from ATR - Interpreting Telecommunications Research Laboratories of Japan, the US Department of Defense, and the Verbmobil Project of the Federal Republic of Germany.

We would like to thank all members of the JANUS teams at the University of Karlsruhe and Carnegie Mellon University for their dedicated work on our many evaluations.

In Domain (605 sentences)

	GLR*		Phoenix	
	transcribed	speech 1st-best	transcribed	speech 1st-best
Perfect	65.2	34.7	53.3	35.5
OK	18.8	12.2	25.3	26.3
Bad	16.0	29.2	21.4	17.1
Recog Err	**	23.9	**	21.1
Out of Domain (485 sentences)				
Perfect	58.5	29.7	44.2	29.3
OK	26.7	42.4	44.6	41.1
Bad	14.8	7.5	11.2	9.1
Recog Err	**	20.4	**	20.5
Acceptable (Perfect + OK)				
In Dom	84.0	46.9	78.6	61.8
Out of Dom	85.2	72.1	88.8	70.4
All Dom	84.5	58.2	82.9	65.5

Figure 4: September 1995 evaluation of GLR* and Phoenix. Cross-grading of 16 dialogues.

In Domain (605 sentences)

	GLR* with Phoenix	
	transcribed	speech 1st-best
Perfect	65.4	39.7
OK	20.8	21.2
Bad	13.8	15.2
Recog Err	**	23.9
Acceptable (Perfect + OK)		
In Dom	86.2	60.9

Figure 5: September 1995 evaluation of GLR* combined with Phoenix. Cross-grading of 16 dialogues.

References

D. Gates, A. Lavie, L. Levin, A. Waibel, M. Gavaldà, L. Mayfield, M. Woszczyna and P. Zhan. *End-to-end Evaluation in JANUS: a Speech-to-speech Translation System,* To appear in Proceedings of ECAI Workshop on Dialogue Processing in Spoken Language Systems, Budapest, Hungary, August 1996.

A. Lavie and M. Tomita. *GLR* - An Efficient Noise Skipping Parsing Algorithm for Context Free Grammars,* Proceedings of the third International Workshop on Parsing Technologies (IWPT-93), Tilburg, The Netherlands, August 1993.

A. Lavie. An Integrated Heuristic Scheme for Partial Parse Evaluation, Proceedings of the 32nd Annual Meeting of the ACL (ACL-94), Las Cruces, New Mexico, June 1994.

L. Mayfield, M. Gavaldà, Y-H. Seo, B. Suhm, W. Ward, A. Waibel. "Parsing Real Input in JANUS: a Concept-Based Approach." In *Proceedings of TMI 95.*

C. P. Rosé, B. Di Eugenio, L. S. Levin, and C. Van Ess-Dykema. Discourse processing of dialogues with multiple threads. In *Proceedings of ACL'95, Boston, MA*, 1995.

M. Tomita. An Efficient Augmented Context-free Parsing Algorithm. *Computational Linguistics*, 13(1-2):31–46, 1987.

M. Tomita. The Generalized LR Parser/Compiler - Version 8.4. In *Proceedings of International Conference on Computational Linguistics (COLING'90)*, pages 59–63, Helsinki, Finland, 1990.

M. Tomita and E. H. Nyberg 3rd. Generation Kit and Transformation Kit, Version 3.2: User's Manual. Technical Report CMU-CMT-88-MEMO, Carnegie Mellon University, Pittsburgh, PA, October 1988.

W. Ward. "Understanding Spontaneous Speech: the Phoenix System." In *Proceedings of ICASSP-91*, 1991.

W. Ward. "Extracting Information in Spontaneous Speech." In *Proceedings of International Conference on Spoken Language*, 1994.

M. Woszczyna, N. Aoki-Waibel, F. D. Buo, N. Coccaro, T. Horiguchi, K. and Kemp, A. Lavie, A. McNair, T. Polzin, I. Rogina, C. P. Rosé, T. Schultz, B. Suhm, M. Tomita, and A. Waibel. JANUS-93: Towards Spontaneous Speech Translation. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'94)*, 1994.