Translation of Conversational Speech with JANUS-II

Alon Lavie, Alex Waibel, Lori Levin, Donna Gates, Marsal Gavaldà,
Torsten Zeppenfeld, Puming Zhan and Oren Glickman

Center for Machine Translation
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213
Email : lavie@cs.cmu.edu

ABSTRACT

In this paper we investigate the possibility of translating continuous spoken conversations in a cross-talk environment. This is a task known to be difficult for human translators due to several factors. It is characterized by rapid and even overlapping turn-taking, a high degree of co-articulation, and fragmentary language. We describe experiments using both push-to-talk as well as cross-talk recording conditions. Our results indicate that conversational speech recognition and translation is possible, even in a free crosstalk environment. To date, our system has achieved performances of over 80% acceptable translations on transcribed input, and over 70% acceptable translations on speech input recognized with a 70-80% word accuracy. The system's performance on spontaneous conversations recorded in a cross-talk environment is shown to be as good and even slightly superior to the simpler and easier push-to-talk scenario.

1. Introduction

Below, we describe the JANUS system [7] and show its application to the problem of the translation of conversational dialogues in a cross-talk environment. Switching the recording conditions from push-to-talk to cross-talk creates several complicating factors, making the task more difficult, yet also more realistic. Conversational speech in a cross-talk environment is characterized by rapid and even overlapping turn-taking, a high degree of co-articulation, and fragmentary language.

We begin with an overview of the JANUS translation system, including a description of the individual modules and their function. We then describe our evaluation methodology, and conclude with a summary of our current results.

A component diagram of our system can be seen in Figure 1. The main system modules are speech recognition, parsing, discourse processing, and generation. Each module is language independent in the sense that it consists of a general processor that can be loaded with language specific knowledge sources. In an attempt to achieve both robustness and translation accuracy when faced with speech disfluencies and recognition errors, we use two different parsing strategies: a GLR parser designed to be more accurate, and a Phoenix parser designed to be more robust.

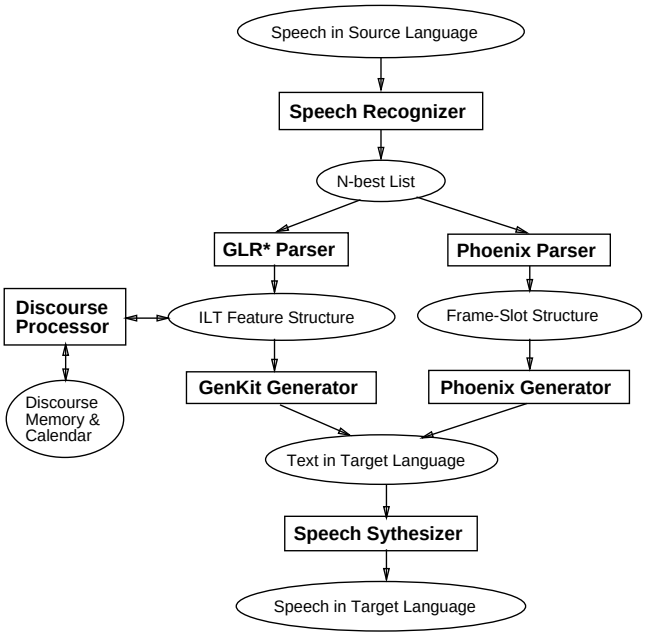

Figure 1: The JANUS System

Speech translation in the JANUS system is guided by the general principle that spoken utterances can be analyzed and translated as a sequential collection of semantic dialogue units (SDUs), each of which roughly corresponds to a speech-act. SDUs are semantically coherent pieces of information. The interlingua representation in our system was designed to capture meaning at the level of such SDUs. Each semantic dialogue unit is analyzed into an interlingua representation. For both parsers, segmentation into SDUs is achieved in a two-stage process, partly prior to and partly during parsing.

In order to efficiently process multiple speech hypotheses, we have adapted our parsers to process speech lattices, which are output by the recognizer. The lattice representation is efficient because common portions of different hypotheses are represented only once in the lattice. This allows the parser to analyze multiple hypotheses for a given input. In order to disambiguate among these multiple hypotheses, our strategy has been to apply a late stage disambiguation,

which utilizes knowledge from all the machine translation components - acoustic and language models, parser scores, and contextual information obtained from discourse analysis. Each of these components provides a score for each possible analysis of an ambiguous input. One current research topic is the development of methods for combining these scores in a way that achieves optimal performance.

2. Speech Recognition

The first main component in our speech-to-speech translation system is the speech recognizer. Its job is to decode the speech of a user and turn it into text to be passed to the parsing/translation modules. Our baseline JANUS-II recognizers use one or more streams of input features derived from Mel-scale or PLP filters processed using Linear Discriminant Analysis (LDA). The acoustic units are context dependent 3-state Triphones, modeled via continuous density HMMs. Explicit noise models are added to help the system cope with breathing, lip-smack, and other human and non-human noises inherent in a spontaneous speech task.

While speech recognition systems readily achieve word accuracies of 90+% on read speech, conversational speech poses a much more difficult problem, and generally results in higher word error rates. Our JANUS-II recognition system has been applied to various conversational speech tasks, and now achieve Word Accuracies of 90+% on the japanese spontaneous scheduling task (JSST), 70-80% on GSST and ESST (German and English respectively), and a WER of 38.4% on the Switchboard LVCSR task.

Some of the recent improvements that have been introduced into our system include:

- **MLLR Codebook Adaptation** - An unsupervised adaptation technique for use in speaker adaptation.
- **Decision tree acoustic model clustering** - A technique to automatically find the appropriate number and placement of parameters in our acoustic models.
- **Dictionary Learning** - Due to the variability, dialect variations, and coarticulation phenomena found in spontaneous speech, pronunciation dictionaries have to be modified and fine-tuned for each language. To eliminate costly manual labor and for better modeling, we resort to data-driven ways of discovering such variants.
- **Morpheme Based Language Models** - For languages characterized by a richer morphology, use of inflections and compounding (compared with English), more suitable units than the 'word' are used for dictionaries and language models.
- **Phrase Based and Class Based Language Models** - Words that belong to word classes (such as days of the week), or frequently occurring phrases (e.g., *out-of-town*, *I'm-gonna-be*, *sometime-in-the-next*) are discovered automatically by clustering techniques and added to a dictionary as special words, phrases or mini-grammars.

3. The Robust GLR and Phoenix Translation Modules

JANUS employs two robust translation modules with complementary strengths. The GLR module gives more complete and accurate translations whereas the Phoenix module is more robust over the disfluencies of spoken language. The two modules can run separately or can be combined to gain the strengths of both.

The GLR module is composed of the GLR* parser [2][3], the LA-Morph morphological analyzer and the GenKit generator. The GLR* parser is based on Tomita's Generalized LR parsing algorithm [5]. GLR* skips parts of the utterance that it cannot incorporate into a well-formed sentence structure. Thus, it is well-suited to domains in which non-grammaticality 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 feasible time and space bounds. JANUS GLR grammars 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 GLR* parser also includes several tools designed to address the difficulties of parsing spontaneous speech, including a statistical disambiguation module, a self-judging parse quality heuristic, and the ability to segment multi-sentence utterances. Target language generation is done using GenKit, a unification-based generation system. With well-developed generation grammars, GenKit results in very accurate translation for well-specified ILTs.

The JANUS Phoenix translation module [4] is an extension of the Phoenix Spoken Language System [6]. It consists of a parsing module and a generation module. 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. The Phoenix parsing grammar specifies patterns which represent concepts in the domain. 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 parser matches as much of the input utterance as it can to the patterns specified by the RTNs. 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 may add to the 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. 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. The result is a meaningful but somewhat telegraphic translation.

Although both GLR* and Phoenix were specifically designed to deal with spontaneous speech, each of the approaches has some clear strengths and weaknesses. 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 Section 6. We are in the process of investigating some more sophisticated methods for combining the two translation approaches.

4. Lattice Parsing

Speech recognition errors hinder the ability of the parser to find a correct analysis for the utterance. This is reflected in the disparity between our performance results on transcribed and speech recognized input. Processing multiple speech hypotheses instead of a single top-best hypothesis has the potential of detecting a hypothesis with fewer recognition errors, which should lead to an improvement in the overall translation performance.

Parsing the speech lattice directly attempts to efficiently accomplish the same results as parsing a list of hypotheses. Each word in the lattice is parsed only once, although it may contribute to many different hypotheses. The lattice parser produces a large set of possible parses of various complete word paths through the lattice. This set of parses can be scored and ranked according to an optimized combination of the parser score and recognizer score.

The lattices produced by our speech recognizer are too large and redundant to be parsed directly. We apply four steps to make them more tractable. The first step involves cleaning the lattice by mapping all non-human noises and pauses into a generic pause. The resulting lattice contains only linguistically meaningful information. The lattice is then broken at points where the speech signal contains long pauses, which are highly indicative of sentence boundaries, yielding a set of sub-lattices. Each of the sub-lattices is then re-scored by the language model. Finally, the lattices are pruned to a size that the parser can process in reasonable time and space. The re-scoring raises the probability that the correct hypothesis will not be lost during the pruning stage. The resulting sub-lattices are sequentially passed on to the parser.

The lattice parsing version of GLR* extended the parser to effectively deal with multiple speech hypotheses represented in the form of a lattice. In order to correctly consider only valid hypotheses in the lattice, the parser uses a procedure for determining the connectivity of two points in the lattice. Enhanced ambiguity packing allows the parser to efficiently represent the collection of sub-parses found for various parts of the lattice. We are also in the process of developing a lattice-parsing version of the Phoenix parser.

5. Late-stage Disambiguation

An important feature of our translation approach is to allow multiple hypotheses to be processed through the system, and to use context to disambiguate between alternatives in the final stage of the process, where knowledge can be exploited to the fullest. Since it is infeasible to process *all* hypotheses produced by each of the system components, context is also used locally to prune out unlikely alternatives. A post-parsing procedure selects the top k packed parses from the list of parses (k is an adjustable constant). These parses will correspond to different paths through the lattice. Each parse is first unpacked and disambiguated. Next, the path of lattice words associated with each of the parses is retrieved and the acoustic score of this path is calculated and attached to the parse. The final disambiguation combines all knowledge sources obtained: the acoustic score, the parser score, and information obtained from the discourse processor [1]. The best scoring hypothesis is then sent to the speech synthesizer. This hypothesis is also sent back to the discourse processor so it can update its internal structures and the discourse state.

Perfect	Fluent translation with all information conveyed
OK	All important information translated correctly but some unimportant details missing or translation is awkward
OK tagged	The sentence or clause is out-of-domain and no translation is given.
Bad	Unacceptable translation

Figure 2: Evaluation Grade Categories

	Transcription	Output of Speech-recognition
GLR*	84.1%	46.9%
Phoenix	78.6%	61.7%
Combined	86.2%	60.9%

Figure 3: End-to-end Translation Performance Results

6. Evaluation Methods and Results

The goal of our evaluation methods is to provide a meaningful and accurate measure of the capability of our system as a whole. We accomplish this by periodically testing our system on sets of "unseen" data. The data chosen for testing consists of dialogues by speakers whose voices were not used for training or development of both the speech recognizer and the translation components. We perform evaluations on the end-to-end system from speech recognition through target language generation. A similar evaluation is conducted using transcribed input instead of speech recognized input. This allows us to isolate performance deficiencies that are solely due to speech recognition errors. The evaluations are scored by an independent grader. We employ a consistent set of criteria for judging the quality of the utterances as well as their relevance to the current domain. Each SDU is assigned a separate grade. A grading assistant program helps the scorer in assigning SDU level scores, tabulates and saves the results. Figure 2 lists the possible grades and the criteria for assigning them. The translation modules attempt to detect out-of-domain SDUs (in this case, SDUs that are not about scheduling meetings) and avoid giving them erroneous translations. An SDU that is recognized as out-of-domain and not translated is given the score "OK tagged".

The results in Figure 3 show the performance of the GLR and the

[1] We are still experimenting with the weights assigned to each of the scores in this combination.

Phoenix Spanish-English translation modules on a recent test set. The test set consisted of 15 dialogues recorded in a cross-talk setting (see following subsection), with a total of 349 utterances. The results shown are for in-domain SDUs only. The numbers reported are the percent of acceptable translations, which is the sum of perfect and OK translations. Results are shown for both transcribed and speech recognized versions of the input, and using either the GLR* or the Phoenix parser. In this evaluation, only the top-best hypothesis of the speech recognizer was used. The speech recognition average word accuracy on this test set was 62.1%. As can be seen, while GLR* achieves better translation results on transcribed data, the Phoenix parser was better in overcoming errors due to speech recognition. The results in the last row of Figure 3 reflect the combination of the GLR* and Phoenix systems as described in Section 3. In a separate evaluation of the lattice processing configuration of the system, we noted about a 3% improvement in end-to-end translations when processing lattices rather than the top-best speech hypothesis [2].

6.1. Comparison of Push-to-talk and Cross-talk Performance

In earlier stages of the project, our speech recordings were conducted in a push-to-talk setting, where each speaker activated the communication (and recording) by explicitly pressing a button while speaking. The two speakers were not allowed to overlap in their conversation, but rather took turns in conversing. We recently decided to experiment with the more challenging cross-talk setting. In the cross-talk setting, the two speakers are recorded on separate channels, but are free to converse in a completely spontaneous fashion, at times cutting into and overlapping the other speaker. Since the level of spontaneity in the cross-talk setting is much higher, we expected to suffer a noticeable degradation in system performance. Figure 4 shows our performance results on the above mentioned cross-talk test set, and also the results on a smaller push-to-talk test set of 41 utterances. Note that it is not possible to directly compare the two columns, because by the nature of the experiment the two data-sets are different. However, we note that our translation performance did not in fact decrease as was expected. If anything, it increased. One possible explanation might be that the shorter utterances in the cross-talk setting are easier to translate, and the parsers succeed in segmenting them more correctly into SDUs.

7. Conclusions and Future Work

In this paper we described the methods we employ to integrate speech recognition and translation in the JANUS system. Taking advantage of the complementary strengths of our two robust parsers allows us to overcome the disfluencies and ungrammaticalities that are typical of spoken language. Our end-to-end evaluation procedures allow us to assess the overall performance of the system, using each of the translations methods separately or both combined. Lattice parsing offers the potential of overcoming many speech recognition errors. However, this requires the development of better methods for pruning the lattices without the loss of the hypothesis with the best word accuracy.

Our current and future research efforts concentrate on improved methods for combining the scores of our different knowledge sources, improving the method by which we combine the two translation engines, and the automatic detection of out-of-domain segments and utterances.

[2] While the complete lattices from the speech recognizer had a word accuracy of 91%, the lattices after pruning in this evaluation had a word accuracy of only 84%. The average word accuracy of the top-best hypothesis on this test set was 80%. This explains why our translation performance improved by only 3%.

	Push-to-talk	Cross-talk
Speech Recognition Word Accuracy	71%	70%
Translation Performance		
GLR* Transcribed data	77%	83%
Phoenix Transcribed data	74%	81%
GLR* SR data	44%	65%
Phoenix SR data	52%	73%

Figure 4: End-to-end Translation Performance on Push-to-talk and Cross-talk Data

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.

8. REFERENCES

1. 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.
2. 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.
3. 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.
4. 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*.
5. M. Tomita. An Efficient Augmented Context-free Parsing Algorithm. *Computational Linguistics*, 13(1-2):31–46, 1987.
6. W. Ward. Extracting Information in Spontaneous Speech. In *Proceedings of International Conference on Spoken Language*, 1994.
7. 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.