JANUS: a Multi-lingual Speech-to-speech Translation System for Spontaneously Spoken Language in a Limited Domain

**Alon Lavie, Lori Levin, Alex Waibel,
Donna Gates, Marsal Gavalda and Laura Mayfield**

Center for Machine Translation
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213
email : lavie@cs.cmu.edu
Phone: +1-412-268-5655
Fax: +1-412-268-6298

Abstract

Janus is a multi-lingual speech translation system currently operating in the domain of meeting scheduling. Translating spontaneous speech requires a high degree of robustness to overcome the disfluencies of spoken language as well as errors in speech recognition. In this system description, we focus on the robust speech translation components in Janus—the skipping GLR* parser, the segmentation of full utterances into semantic dialogue units (SDUs), and the late-stage disambiguation of utterances. We will also describe how the end-to-end translation performance of the system is evaluated and present our latest Spanish-to-English evaluation results.

System Description

JANUS: a Multi-lingual Speech-to-speech Translation System for Spontaneously Spoken Language in a Limited Domain

Alon Lavie, Lori Levin, Alex Waibel,
Donna Gates, Marsal Gavalda and Laura Mayfield
Center for Machine Translation
Carnegie Mellon University

1 Introduction

Janus is a multi-lingual speech translation system currently operating in the domain of meeting scheduling. Translating spontaneous speech requires a high degree of robustness to overcome the disfluencies of spoken language as well as errors in speech recognition. In this system description, we focus on the robust speech translation components in Janus—the skipping GLR* parser, the segmentation of full utterances into semantic dialogue units (SDUs), and the late-stage disambiguation of utterances. We will also describe how the end-to-end translation performance of the system is evaluated and present our latest Spanish-to-English evaluation results.

2 Robust Parsing of Spontaneous Speech

The GLR* parser [Lavie and Tomita 1993, Lavie 1994] is a parsing system based on Tomita's Generalized LR parsing algorithm [Tomita 1987], specifically designed to robustly handle spontaneously spoken language. The parser skips parts of the utterance that it cannot incorporate into a well-formed sentence structure. 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]. For the scheduling domain, we use semantic grammars, in which the grammar rules define semantic categories such as `busy-free-p` and `schedule-meeting` in addition to syntactic categories such as `NP` and `VP`.

The semantic 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 1. 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. The GenKit generation module [Tomita and Nyberg 1988] is used to convert ILTs into target language text.

```
((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 1: An Example ILT

3 Segmentation of Full Utterances

Full utterances in Janus are treated as a sequence of Semantic Dialogue Units (SDUs), which each correspond roughly to a speech act. Often, SDUs are not complete grammatical sentences. The analysis grammars are designed to map each SDU onto an interlingual text (ILT). The analysis of a full utterance into SDUs requires the ability to correctly identify boundaries between units. Utterance segmentation is performed partly prior to parsing and partly during analysis by the parser. Figure 2 shows an example of how a full utterance is segmented into SDUs.

Pre-parsing segmentation relies on acoustic, lexical, syntactic, semantic, and statistical knowledge sources. Acoustic cues that have a high probability of occuring at SDU boundaries are silences, two or more human noises in a row, or three or more non-human noises in a row. The second source of information for pre-parsing segmentation is a statistical measure that attempts to capture the likelihood of an SDU boundary between any two words of an utterance. The measure is trained on hand-segmented transcriptions of dialogues. The third source of information for pre-parsing segmentation is lexical cues. For example, the phrases *qué tal*, *qué te parece*, and *si* usually occur after an SDU boundary while *Sí* and *claro* occur before an SDU boundary. The advantages of pre-parsing segmentation are reduction in parsing time, increase in parse accuracy, and reduction in ambiguity.

Pre-parsing segmentation may not identify all SDU boundaries. Each segment returned by the pre-parsing procedures can be further segmented by the parser, which can make use of syntactic and semantic constraints. Three methods are applied here. First, the grammar rules apply penalties to parses that are fragmented. Less fragmented analyses are preferred. Second, the parser makes use of the same statistical measure used during pre-parsing segmentation, but instead of using it to predict likely SDU boundaries, the parser uses it to prevent the consideration of an SDU boundary at unlikely locations in the utterance. Third, the parser's disambiguation procedures are responsible for picking the best analysis of all those produced so far. The parser's disambiguation procedures take into account the amount of skipping, the amount of fragmentation, and the probabilities of shift and reduce actions taken during each parse.

The overall test of our segmentation procedures is whether they result in an improvement in translation of spoken dialogues. Our initial pre-parsing method for segmenting utterances relied only on acoustic cues. The addition of lexical cues and the statistical measure resulted in an improvement in acceptable translations on in-domain SDUs from 51% to 61% (with a speech recognition word accuracy of 68%). Furthermore, parsing time was significantly reduced.

```
Unsegmented Speech Recognition:

  (%noise% si1 mira toda la man5ana estoy disponible %noise% %noise% y tambie1n el fin de semana
   si podri1a hacer mejor un di1a fin de semana porque justo el once no puedo me es imposible
   va a poder fin de semana %noise%)

Pre-broken Speech Recognition:

  (si1)
  (mira toda la man5ana estoy disponible %noise% %noise% y tambie1n el fin de semana)
  (si podri1a hacer mejor un di1a fin de semana)
  (porque justo el once no puedo me es imposible va a poder fin de semana)

Parser SDU Segmentation (of Pre-broken Input):

 (((si1))
  ((mira) (toda la man5ana estoy disponible) (y tambie1n) (el fin de semana))
  ((si podri1a hacer mejor un di1a fin de semana))
  ((porque el once no puedo) (me es imposible) (va a poder fin de semana)))

Translation:

  "yes --- Look all morning is good for me -- and also -- the weekend ---
   If a day weekend is better --- because on the eleventh I can't meet --
   That is bad for me can meet on weekend"
```

Figure 2: SDU Segmentation of a Spanish Full Utterance

4 Parse Disambiguation

Resolution of ambiguity is important for accurate translation. The approach we have taken is to allow multiple hypotheses and their corresponding ambiguities to cascade through the translation components, accumulating information that is relevant to disambiguation along the way. In contrast to other approaches that use predictions to filter out ambiguities early on, we delay ambiguity resolution as much as possible until the stage at which all knowledge sources can be exploited. A consequence of this approach is that much of our research effort is devoted to the development of an integrated set of disambiguation methods that make use of statistical and symbolic knowledge.

Disambiguation in GLR* 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. Each evaluation measure is a penalty function, which assigns a penalty score to each of the alternative analyses, according to its desirability. The parser currently combines three penalty scores: (1) A penalty for skipping words that takes into account their saliency in the domain, (2) a penalty corresponding to the fragmentation counter assigned by the grammar rules, and (3) A penalty based on the probabilities of shift and reduce actions in the LR parsing table. The penalty scores are then combined into a single score using a linear combination. A parse quality heuristic allows the parser to self-judge the quality of the parse chosen as best, and to detect cases in which important information is likely to have been skipped. Additional penalty scores are later assigned by the discourse processor, which is also responsible for effectively combining its additional scores with the parser scores. We are currently experimenting with some non-linear approaches to combining discourse and parser scores.

	Transcribed	Speech 1st-best
November-94	72.2 %	49.2 %
April-95	81.9 %	55.3 %
September-95	84.5 %	58.2 %

Figure 3: Improvement in Percentage of Acceptable Translations over Time

5 Evaluation of End-to-End Performance

We regularly evaluate the performance of the Janus system on an end-to-end basis—spoken source language input to written target language output. Evaluations are performed to verify the coverage of our knowledge sources, guide our development efforts, and track our progress over time. It is essential that the evaluations be performed on unseen data that reflects translation performance under real conditions. The evaluation procedure also (1) employs a set of consistent criteria for judging, but is also designed to compensate for subjectivity in scoring (2) takes into account utterance complexity, and (3) compensates for data that is not relevant to the domain being evaluated.

Evaluations are performed on sets of unseen data averaging about one hundred utterances. System performance is first evaluated prior to any development on the data. After the first evaluation, the system is improved to cover deficiencies and then re-evaluated to determine the effect of development on performance. To track our performance over time, several backed-up versions of the system are tested on a single set of unseen data. A table showing improvement over three system versions of our Spanish-to-English translation is shown in Figure 3.

Rather than assigning one score to each utterance, we assign a separate score to each SDU. This gives more weight to long utterances and allows us to more accurately judge utterances that contain both in-domain and out-of-domain information. Our main measure of translation quality is the number of *acceptable* translations, which is the sum of the number of *Perfect* and *OK* translations.

References

[Lavie and Tomita 1993] 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.

[Lavie 1994] 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.

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

[Tomita 1990] 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.

[Tomita and Nyberg 1988] 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.