JANUS: A SPEECH-TO-SPEECH TRANSLATION SYSTEM USING CONNECTIONIST AND SYMBOLIC PROCESSING STRATEGIES

Alex Waibel *Ajay N. Jain* *Arthur E. McNair*
Hiroaki Saito *Alexander G. Hauptmann* *Joe Tebelskis*

School of Computer Science, Carnegie Mellon University
Pittsburgh, Pennsylvania 15213-3890, USA

ABSTRACT

We present JANUS, a speech-to-speech translation system that utilizes diverse processing strategies including dynamic programming, stochastic techniques, connectionist learning, and traditional AI knowledge representation approaches. JANUS translates continuously spoken English utterances into Japanese and German speech utterances. The overall system performance on a corpus of conference registration conversations is 87%. Two versions of JANUS are compared: one using an LR parser (JANUS1) and one using a connectionist parser (JANUS2). Performance results are mixed, with JANUS1 deriving benefit from a tighter language model and JANUS2 benefitting from greater flexibility.

1. INTRODUCTION

In an age of increasing internationalization, efficient and rapid communication between people around the world has become a necessity of modern business life. It has created a rising need for better communication tools that could help bridge the language and culture gaps that still isolate one people from another. In this spirit, automated speech translation has become a research target for a number of research laboratories (e.g. [1]). Real-time translation of telephone conversations is an ambitious project. It requires the integration of three component technologies: speech recognition, machine translation, and speech synthesis. Each of these technologies is currently under active research.

In this paper, we present JANUS [1], a speech-to-speech translation system developed at Carnegie Mellon. JANUS is able to translate continuously spoken English speech utterances into Japanese and German speech utterances. The system consists of several sub-components that utilize different processing strategies ranging from connectionist systems to LR parsing algorithms (see Fig. 1). JANUS currently operates on a conference registration dialog task. In the conversations, a caller is attempting to obtain information or to register for an international conference by interacting with a conference secretary. The speech dialogs were acted (read) out under benign recording conditions.

As a speech recognition front-end, we use a continuous speech recognition system based on connectionist acoustic modeling (LPNNs) and stochastic language modeling techniques (bigram grammars). At the language processing and translation level, we describe and evaluate two modules using contrasting computational techniques—knowledge-based versus connectionist language processing. Speech synthesis, finally, is performed by two commercial text-to-speech synthesis devices, one for German and one for Japanese. In the evaluations that follow, we consider semantic fidelity, rather than lexical description accuracy alone. The system's ability to deliver the correct translation in the face of potential recognition errors is our primary goal. A long-term goal is to demonstrate tolerance to speech effects such as ungrammaticality, stuttering, interjections, etc.

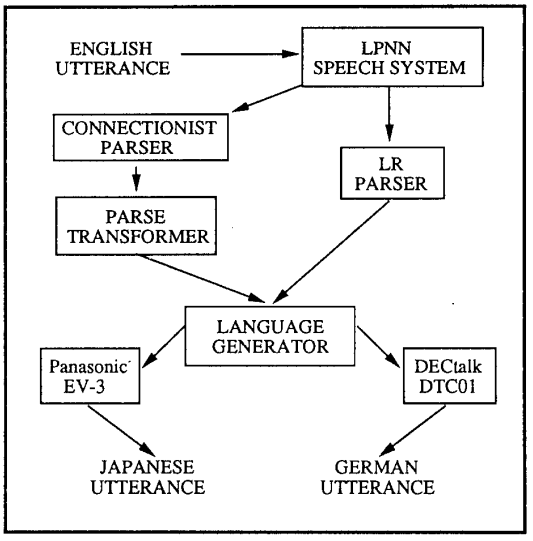

Figure 1: JANUS system components.

In what follows, we present the structure and performance of the JANUS system. First, we describe the speech recognition and synthesis components. Next, we detail the structure of the machine translation system and its interchangeable parsing components—a generalized LR parser and a connectionist parser. Lastly, we compare the performance of the system using these different components.

[1] The system is named after the Roman god with two faces.

CH2977-7/91/0000-0793 $1.00 © 1991 IEEE

```
LPNN output:
(HELLO IS THIS THE OFFICE FOR
  THE CONFERENCE $)

Parser's interlingual output:
((CFNAME *IS-THIS-PHONE)
 (MOOD *INTERROGATIVE)
 (OBJECT ((NUMBER SG)
  (DET THE)
  (CFNAME *CONF-OFFICE)))
 (SADJUNCT1
  ((CFNAME *HELLO))))


Japanese translation:
MOSHI MOSHI KAIGI JIMUKYOKU DESUKA

German translation:
HALLO IST DIES DAS KONFERENZBUERO
```

Figure 2: JANUS using the generalized LR parser.

```
LPNN output:
(HELLO IS THIS THE OFFICE FOR
  THE CONFERENCE $)

Connectionist parse:
((QUESTION 0.9)
 ((GREETING 0.8)
   ((MISC 0.9)     HELLO))
 ((MAIN-CLAUSE 0.9)
   ((ACTION 0.9)   IS)
   ((AGENT 0.9)    THIS)
   ((PATIENT 0.8) THE OFFICE)
   ((MOD-1 0.9)    FOR THE CONFERENCE)))

Japanese translation:
MOSHI MOSHI KAIGI JIMUKYOKU DESUKA

German translation:
HALLO IST DIES DAS KONFERENZBUERO
```

Figure 3: JANUS using the connectionist parser.

2. SPEECH RECOGNITION AND SYNTHESIS

Speech recognition in the JANUS system is provided by a connectionist, continuous, large vocabulary, Linked Predictive Neural Network (LPNN) system developed by Tebelskis [2]. This system, as used in JANUS, is speaker dependent, has a vocabulary of 400 English words (sufficient for the conference registration task), and uses a statistical bigram grammar of perplexity 5. The LPNN module can produce either a single hypothesized textual sentence or the first N best hypotheses. To produce the N best hypotheses, a modified dynamic-programming beam-search algorithm is used, similar to[3]. This system, when using the bigram grammar, produces the correct sentence as one of the top 3 choices in 90% of the cases, with additional gains within the top 9 choices. The text output from the recognizer is processed by the next component of JANUS, the parsing/translation module, which will be described later.

Speech synthesis is provided by two commercially available devices, a Digital DECtalk DTC01 system for German output, and the Panasonic Text-to-Speech System EV-3 for Japanese output. Each of these systems takes a textual or phonetic representation of a sentence as input, and produces the sounds of the spoken utterance through an audio speaker. The following two sections describe the alternative parsing/translation modules.

3. KNOWLEDGE BASED MACHINE TRANSLATION

The first parsing/translation module is the Universal Parser Architecture (UPA) developed at Carnegie Mellon[4]. It is a knowledge-based machine translation system that is capable of performing efficient multi-lingual translation. The system consists of a parsing component and a generation component. The parsing component makes use of Tomita's efficient generalized LR parsing algorithm[5]. After pre-compilation of a grammar, fast table-lookup operations are all that is necessary to parse utterances. The performance of this module approaches real-time. Language generation also approaches real-time and is performed using GenKit, a system that compiles a generation grammar into LISP functions[6].

The UPA system requires a hand-written grammar for each language to be used for parsing and generation. The system uses a Lexical Functional Grammar formalism, and both syntactic and semantic rules are encoded in the grammar. Multi-lingual parsing is achieved by writing grammars for each of several languages. The universal parser with its precompiled grammar takes a text input of a sentence in the source language and produces an "interlingual representation"—a language-independent frame-based representation of the meaning of the input sentence. The universal generator takes this as input, and uses the generation grammar to make the transformation into the appropriate text in the target language. Figure 2 shows an example of the input, interlingual representation, and the output of the UPA system.

4. CONNECTIONIST PARSING

An alternative to the approach to parsing described above is a connectionist parser developed by Jain[7]. This parsing system learns to parse sentences one word at a time from a corpus of training examples. In previous work, we showed that the connectionist parser has three main computational strengths in the domain of speech understanding. First, it learns and generalizes from training examples. This eliminates the need to construct grammars by hand. This task can be especially difficult in spoken domains where grammatical regularity is lacking. Second, by virtue of the learning algorithms that it employs, the parser can effectively combine symbolic information (e.g. syntactic features of words) and non-symbolic information (e.g. statistical likelihood of sentence types). Lastly, we showed that the parser was tolerant of some types of noisy input as might arise

Correct recognition and translation	176	178	87.3%
Incorrect recognition but correct translation	2		
Incorrect recognition and incorrect translation	12	26	12.7%
Incorrect recognition and no parsable utterance	14		

Table 1: Performance of JANUS using the generalized LR parser on all 12 conversations.

Correct recognition and translation	N: 76 F: 65	N: 78 F: 67	N: 89.7% F: 77.0%
Incorrect recognition but correct translation	N: 2 F: 2		
Incorrect recognition and incorrect translation	N: 6 F: 5	N: 9 F: 20	N: 10.3% F: 23.0%
Incorrect recognition and no parsable utterance	N: 3 F: 15		

Table 2: Performance of JANUS using the generalized LR parser on the first 6 conversations (**JANUS1**).

Correct recognition and translation	N: 66 F: 63	N: 72 F: 68	N: 82.8% F: 78.2%
Incorrect recognition but correct translation	N: 6 F: 5		
Incorrect recognition and incorrect translation	N: 14 F: 12	N: 15 F: 19	N: 17.2% F: 21.8%
Incorrect recognition and no parsable utterance	N: 1 F: 7		

Table 3: Performance of JANUS using the connectionist parser on the first 6 conversations (**JANUS2**).

from speech recognition errors or from ungrammatical speech.

For the JANUS system, we have extended the parser to handle a wider variety of sentences. The parser was trained on the same sentences that were available to the human grammar-writer in the UPA system. The parser was constructed from separate connectionist modules arranged in a hierarchical fashion.[2] In the connectionist parsing network, words are represented as binary (primarily syntactic) feature patterns.[3] A word is presented to the network by stimulating the input units to produce the proper pattern for the word. The network makes a series of transformations to the input as it is received. The input word sequence is broken up into phrases, mapped into individual clauses, and the constituents of the structure are assigned labels indicating their function and/or relationship to other constituents. While the parser itself has no failure condition, a few simple heuristics incorporating threshold values on key output units of the connectionist network allow failure detection in many cases.

Figure 3 shows an example of the LPNN output, connectionist parser's output, and the final output of JANUS using the connectionist language component (note that the translations are the same as for the UPA system). The connectionist parser's output is not suitable for direct processing by the language generation module. Transformation of the connectionist parser's primarily syntactic output into the more semantic interlingual representation required by the generation module is accomplished by a separate program. It operates top-down using simple match rules to instantiate case-frames and their slots. The slots of the case-frames are then filled using more match rules. The algorithm is opportunistic in that it attempts to create a reasonable interlingual output representation from any input. Occasionally, the interlingual representation will cause the language generation module to produce a nil output. This is reported as a parsing failure. The following section discusses the overall performance of JANUS as well as the performance of JANUS using the two different parsing strategies.

5. PERFORMANCE

Table 1 shows the performance of JANUS using the UPA parsing/translation component (JANUS1) on the full database of 12 conversations using the N-best utterance hypotheses from the LPNN system.[4] The system returns a translation of the first parsable utterance (or failure in the case of no parsable utterances). The overall performance is 87.3% correct translation. This number includes a small number of cases in which the first parsable utterance was not the actual utterance but produced the correct translation. The 12.7% of the cases where JANUS failed are almost evenly split between two cases. One is where JANUS finds a parsable utterance whose meaning is different from the actual utterance. The other case is where JANUS finds no parsable utterances.

Table 2 shows the performance of JANUS1 on the first 6 conversations in two modes.[5] One mode was where the LPNN produced the N-best utterance hypotheses (labeled "N" in the table). The second mode was where the LPNN produced only the first best hypothesis (labeled "F" in the table). The N-best performance on the first 6 conversations was very similar to the performance for all 12 conversations. The First-best performance was substantially worse than the N-best performance, as expected.

Table 3 shows the performance of JANUS using the connectionist parsing module (JANUS2) on the first 6 conversations. The N-best performance is worse than that for JANUS1. JANUS2 tends to parse and translate utterances that JANUS1 rejects as unparsable. Thus JANUS1, with its tighter language model, has a performance edge. However, the performance of

[2]For a detailed exposition of the connectionist parsing system, see [7].

[3]These feature patterns are not learned by the network, but connectionist networks have been used to acquire such features successfully [8]. From an efficiency perspective, it makes sense to precompile as much lexical information as possible into a network. This is especially important if one does not have a surfeit of training data.

[4]In this and all other performance results, in N-best mode, the 9 best utterance hypotheses are used, i.e. N = 9.

[5]Due to time constraints, the direct comparison of the JANUS1 and JANUS2 was completed for only the first 6 conversations.

JANUS2 in First-best mode does not degrade nearly as much as for JANUS1. In fact, JANUS2 slightly outperforms JANUS1 in First-best mode.

In First-best mode, JANUS1 gets more utterances correct in recognition and translation, but JANUS2 outperforms JANUS1 by more often producing correct translations when the LPNN hypothesis is incorrect. The major difference between JANUS1 and JANUS2 is that JANUS2 (with the connectionist parser) is more likely to successfully parse an utterance that does not correspond to the actual spoken utterance. This is reflected in several ways in Tables 2 and 3. JANUS1 seldom reports parsing failure in N-best mode, but it often does in First-best mode. This means that, when forced to use an imperfect utterance, JANUS1 is more likely to fail to parse it than is JANUS2. JANUS2 reports parsing failure about half as often—the connectionist parser is able to "make do" with imperfect utterances.

The other side to this behavior occurs in N-best mode. The flexibility of the parser in JANUS2 produces a performance loss because it sometimes does not wait long enough to receive the correct utterance. Perhaps if the connectionist parser had a more stringent heuristic for detecting a bad parse, JANUS2 would more closely parallel the performance of JANUS1 in N-best mode.

We have run some experiments comparing the performance of the connectionist parser of JANUS2 with the LR parser of JANUS1 on 117 sentences that were not part of the 12 conversations. For these novel sentences, the connectionist parser substantially outperforms the LR parser. The grammar learned by the connectionist parser has greater coverage than the hand-written grammar of the LR parser. The more extensive coverage of the connectionist parser also accounts for some of the performance difference between JANUS1 and JANUS2 in N-best mode (see above). Note that the grammar for the LR parser was written primarily for correct input and no error correction functions were incorporated. It is possible to write grammars that incorporate such techniques, thus allowing them to handle more varied input[9].

Errors in speech recognition are the primary cause of incorrect translations. Currently, a number of improvements to the speech recognition component are being evaluated. These range from enhancements at the acoustic level to better language modeling[2].

6. CONCLUSION

We have presented a speech-to-speech translation system that utilizes many processing strategies including dynamic programming, stochastic techniques, connectionist learning, and traditional AI knowledge representation. The overall system performance on a set of conference registration conversations was close to 90%. Two versions of JANUS were compared: one using an LR parser (JANUS1) and one using a connectionist parser (JANUS2). The comparison of system performance produced an interesting result. While JANUS1 produced better performance than JANUS2 when allowed multiple utterance hypotheses, JANUS2 performed better than JANUS1 when allowed only a single hypothesis. The connectionist parser showed more tolerance to variations in the structure of its inputs. JANUS2 was more likely to produce a correct translation of an incorrect speech recognition result than JANUS1. Future research will focus on how best to combine the various available technologies to produce the best overall system performance. We are especially interested in understanding how to blend the technologies to handle spontaneous speech effects that are problematic for current systems.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of DARPA, the National Science Foundation, ATR Interpreting Telephony Research Laboratories, NEC Corporation, and Siemens Corporation.

REFERENCES

[1] T. Morimoto, H. Iida, A. Kurematsu, K. Shikano, and T. Aizawa. Spoken language: Towards realizing an automatic telephone interpretation. In *Proceedings of INFO JAPAN 90: International Conference of the Information Processing Society of Japan*, pages 553–559, 1990.

[2] J. Tebelskis, A. H. Waibel, B. Petek, and O. Schmidbauer. Continuous speech recognition using linked predictive neural networks. In *Proceedings of the 1991 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1991.

[3] V. Steinbiss. Sentence-hypotheses generation in a continuous-speech recognition system. In *Proceedings of the 1989 European Conference on Speech Communication and Technology*, volume 2, pages 51–54, 1989.

[4] M. Tomita and J. Carbonell. The universal parser architecture for knowledge-based machine translation. Technical Report CMU-CMT-87-101, Center for Machine Translation, Carneige Mellon University, 1987.

[5] M. Tomita. *Efficient Parsing for Natural Language: A Fast Algorithm for Practical Systems*. Kluwer Academic Publishers, Boston, MA, 1985.

[6] M. Tomita and E. Nyberg. Generation kit and transformation kit. Technical Report CMU-CMT-88-MEMO, Center for Machine Translation, Carneige Mellon University, 1988.

[7] A. N. Jain. Parsing complex sentences with structured connectionist networks. *Neural Computation*, 3:110–120, 1990.

[8] R. Miikkulainen and M. G. Dyer. Encoding input/output representations in connectionist cognitive systems. In D. Touretzky, G. Hinton, and T. Sejnowski, editors, *Proceedings of the 1988 Connectionist Models Summer School*, pages 347–356, Morgan Kaufmann, San Mateo, CA, 1989.

[9] H. Saito and M. Tomita. Parsing noisy sentences. In *Proceedings of the 13th International Conference on Computational Linguistics*, 1988.