A Modular Approach to Spoken Language Translation for Large Domains

Monika Woszczyna, Matthew Broadhead, Donna Gates, Marsal Gavaldà, Alon Lavie, Lori Levin, and Alex Waibel

Interactive Systems Laboratories
at Carnegie Mellon University, Pittsburgh, USA
and Karlsruhe University, Karlsruhe, Germany
monika@cs.cmu.edu

Abstract. The MT engine of the Janus speech-to-speech translation system is designed around four main principles: 1) an **interlingua approach** that allows the efficient addition of new languages, 2) the use of **semantic grammars** that yield low cost high quality translations for limited domains, 3) **modular grammars** that support easy expansion into new domains, and 4) **efficient integration of multiple grammars** using multi-domain parse lattices and domain re-scoring. Within the framework of the C-STAR-II speech-to-speech translation effort, these principles are tested against the challenge of providing translation for a number of domains and language pairs with the additional restriction of a common interchange format.

1 Introduction

Within the Janus project [9] we have been involved in an ongoing effort to develop a machine translation system specifically suited for spoken dialogue. Spoken language is characterized by highly disfluent utterances that are often fragmented and ungrammatical. Furthermore, many communicative acts such as making a polite request involve language specific formulaic expressions. Literal translation of such utterances may not effectively convey the underlying communicative intentions of the speaker. Effective translation of spoken language must therefore be robust and capable of identifying and translating the key underlying concepts of the speaker.

In the most recent version of Janus, our focus has been on extending the capabilities of the system to handle large and rich domains. Within the framework of the C-STAR-II speech-to-speech translation effort[1], we have been developing a translation system for the broad domain of travel planning, which contains a rich structure of sub-domains. Figure 1 shows the complete set of input and output languages that are covered by the C-STAR-II translation effort.

[1] C-STAR is the Consortium for Speech Translation Advanced Research. The C-STAR-II partners are: ATR, Japan; ISL, Universität Karlsruhe, Germany; ISL, Carnegie Mellon University, USA; ETRI, Korea; IRST, Italy; CLIPS-GETA, France. See http://www.is.cs.cmu.edu/cstar

Published in "Proceedings of AMTA-98"

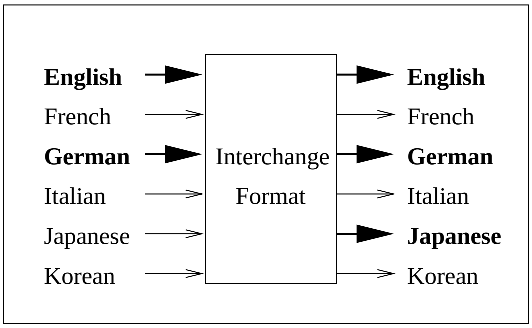

Fig. 1. C-STAR II Languages. Input and output languages analyzed by our group are marked in bold face.

The current JANUS MT system is designed around the following four main principles:

An Interlingua-based Approach: Following an interlingua-based approach allows us to easily expand our system to new languages. Since each language is usually integrated as both a source and a target language, input analyzed into an interlingua representation can then be translated back into the same language as a paraphrase of the original input utterance. This enables a user who has no knowledge of the target language to assess if the system correctly analyzed the input utterance. In our earlier work on the appointment scheduling domain (SST), the SOUP parser [5] generated an interlingua text (ILT), from which the output sentence was generated. For the translation of travel dialogues in the C-STAR project, a common interchange format (IF) for all six member sites was defined. Even though this IF differs considerably from the ILT we used for the scheduling task, our current system architecture can integrate them in a unified system as explained in more detail below.

Semantic Grammars: Semantic grammars have been shown to be effective in providing accurate translation for limited domains. They are also known to be robust against ungrammaticalities in spontaneous speech and recognition errors in speech-to-speech translation systems [6],[7],[10]. However, they are usually hard to expand to cover new domains.

Modular Semantic Grammars: In our current version of the JANUS system we have developed *modular* grammars to overcome the problems associated with expanding semantic grammars to new domains. Each sub-grammar covers the dialogue acts required for one sub-domain. An additional grammar provides cross-domain dialogue acts such as common openings and closings. All grammars share one library with common concepts, such as time expressions and proper names.

Efficient Integration of Multiple Grammars: Our current system is designed to integrate multiple domain grammars in a common analysis module. The parser analyzes the input with multiple grammars concurrently. Analyzed segments of the input are tagged with an ID that reflects the domain grammar that was used in creating the analysis. Segmentation of long utterances into a sequence of DAs is performed as part of the parsing process. The parser produces a lattice output of all possible parsable segments according to the different domain grammars. A statistical domain re-scoring procedure is then applied to the lattice, in order to reduce the level of ambiguity that arises from the combination of multiple domain grammars.

These main themes are described in greater length in the remaining sections of the paper. In Section 2 we describe the interchange format that has been developed for the C-STAR-II multi-site translation effort. In Section 3 we detail our approach for constructing modular semantic grammars which are used to extend our system coverage to new domains. We also describe in detail how these grammars are then integrated together in the runtime architecture of our translation system. In the final section we present a preview of current work on alternative backup methods for translation.

2 The C-STAR Interchange-Format

While the Janus project has always used an interlingua for translation among multiple languages, the C-STAR project presents a special challenge by requiring an interlingua to be used at multiple research sites. It was therefore necessary to design an simple interlingua that could be used reliably by many MT developers. Simplicity is possible largely because we are working on travel planning, a task-oriented domain. In a task-oriented domain, most utterances perform a limited number of *domain actions* (DAs) such as requesting information about the availability of a hotel or giving information about the price of a hotel. These domain actions form the basis of the C-STAR interlingua, which is known as the *interchange format*, or IF. The IF does not represent the literal meaning of an utterance and is far-removed from the source language syntax. It represents only the domain action that the utterance was intended to perform. Translation via a shallow DA-based interlingua is also used in the Verbmobil project, although there it complements a transfer approach which is based on deeper semantic representations [1].

The design principles of the IF are 1) that it is based on domain actions, 2) that it is compositional, i.e. domain actions are built from an inventory of speech acts, concepts, and arguments, and 3) that it is intended to be suitable for all C-STAR languages.

A DA consists of three representational levels: the *speech act*, the *concepts*, and the *arguments*. In addition, each DA is preceded by a speaker tag (`a:` for agent or `c:` for customer) which indicates who is speaking. Plus signs ("+") separate speech acts from concepts and concepts from each other. In general the speech act and speaker information are obligatory whereas the concepts and the arguments are optional. DAs can be roughly characterized as shown

Published in "Proceedings of AMTA-98"

in (1). However, there are constraints on the order of concepts so that not all combinations are possible.

(1) $speaker : speech\ act\ +concept^{*}\ (argument^{*})$

Examples (2) (3) (4) demonstrate specific DAs that are constructed according to this scheme. In example (2) the speech act is `give-information`, the concepts are `availability` and `room`, and the arguments are `time` and `room-type`. The arguments are inherited through a hierarchy of speech acts and concepts. In this case `time` is an argument of `availability` and `room-type` is an argument of room. Example (3) shows a DA which consists of a speech act with no concepts attached to it. The argument `time` is inherited from the speech act `closing`. Finally, example (4) demonstrates a case of DA which contains neither concepts nor arguments.

(2) On the twelfth we have a single and a double available.
`a:give-information+availability+room`
`(room-type=(single & double),time=(md12))`
(3) And we'll see you on February twelfth.
`a:closing (time=(february, md12))`
(4) Thank you very much
`c:thank`

The DAs in the above examples do not capture all of the information present in their corresponding utterances. For instance they do not represent definiteness, grammatical relations, plurality, modality, or the presence of embedded clauses. These features are generally part of the formulaic, conventional ways of expressing the DAs in English. Their syntactic form is not relevant for translation; it only indirectly contributes to the identification of the DA.

Example (5) shows the English paraphrase, German translation, and Japanese translation for sentence (2).

(5) Input: On the twelfth we have a single and a double available.
Paraphrase: A single and a double room will be available the twelfth.
German: Es gibt Einzelzimmer und Doppelzimmer am zwölften.
Japanese: １２日でしたらシングル・ダブルのどちらも空きがございます。

3 Modular Semantic Grammars

For both analysis and generation we have been developing semantic grammars. Rather than focusing on the syntactic structure of the input, semantic grammars directly describe how surface expressions reflect the underlying semantic concepts that are being conveyed by the speaker. Because they focus on identifying a set of predefined semantic concepts, they are relatively well suited to handle the types of meaningful but ungrammatical disfluencies that are typical

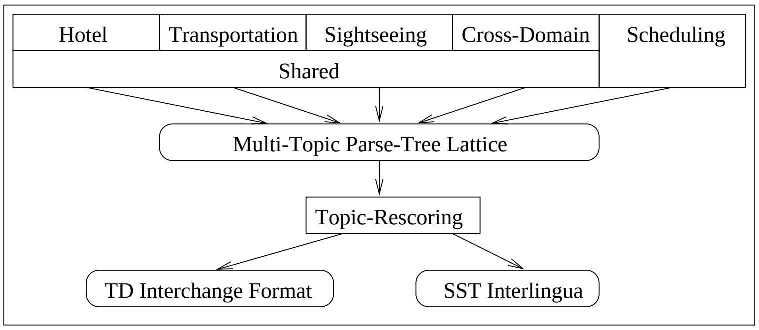

Fig. 2. Combining multiple sub-domain grammars with shared and cross domain grammars.

of spoken language, and are also less sensitive to speech recognition errors. Semantic grammars are also relatively fast to develop for limited domains, where the set of concepts being described is relatively small. However, they are usually hard to expand to cover new domains. New rules are required for each new semantic concept, since syntactic generalities cannot usually be fully utilized. For large domains, this can result in very cumbersome grammars that are difficult to expand and further develop, and which become highly ambiguous in nature.

Modularization and common libraries have long been a well-established concept in software development. Many of the advantages of modularity and shared libraries equally apply to the design of a large semantic grammar for a large domain, particularly if the domain can be dissected into multiple sub-domains. The application of these principles to the development of semantic grammars has the following advantages:

– Separating grammars for sub-domains into independent files allows several grammar developers to work independently and simultaneously without interfering with each other's grammar development.
– The sub-domain grammars draw from a shared library of rules in order to maintain consistency in the analysis of entities such as time and date phrases, auxiliary verbs, etc. The shared library and the cross-domain subgrammar substantially reduce the effort required to expand the system to new domains.
– Separating grammar rules for different sub-domains enables the parser to tag parses of sub-utterances with the corresponding sub-domain. These tags can be used to re-score a lattice of parse trees using conditional probabilities to reduce the ambiguity introduced by expanding to new domains. Lattice re-scoring is explained in more detail in Section 3.3.

3.1 Integration of Multiple Sub-domain Grammars

Figure 2 shows the current configuration used for expanding the grammars to cover a variety of sub-grammars of the travel domain. The SOUP parser reads one sub-domain grammar at a time, and tags the concepts of each grammar with a domain tag, such as HTL (for hotel reservation), TPT (for transportation) and GTR (for general travel), in order to eliminate the possibility of conflicting concept names. All concepts in the shared grammar are left untagged so that they are accessible to all sub-domain grammars.

Since each utterance is parsed as a sequence of DAs, the parser also provides the segmentation of the utterance into DAs. Thus we do not need a separate program for segmenting spoken utterances into sentences. The DAs that comprise one utterance do not have to be taken from the same sub-grammar. The utterance in example (6) contains sub-parses from three different grammars.

(6) Hello,
I would like to make a reservation for a flight to Frankfurt on the fifth and maybe also book a hotel room.
`(GTR) c:greeting`
`(TPT) c:request-action+reservation+temporal+flight`
`(HTL) c:request-action+reservation+features+room`

A considerable advantage of this approach is that grammars producing different interlingua representations can be combined into one system on the sub-utterance level, as shown in example (7), which uses IF and ILT in one utterance. This is possible because in this case the parser is working with a joint grammar that consists of non-overlapping domain-grammars. The output is a sequence of parse-trees, one for each DA in the utterance. Since each parse-tree is marked with a domain ID, it is easy to make sure that each parse-tree is handled by the appropriate mappers and generators.

(7) I would like to make a reservation for a hotel room – do you have time on Friday?
`(HTL) c:request-action+reservation+features+room`
`(SST) q_your_availability`

3.2 Cross Domain and Shared Grammars

The goal of cross domain and shared grammars is to cover the overlap between the grammars for different sub-domains. The cross domain grammar contains grammar rules for dialogue acts that are required in a large number of different tasks. Examples for IF dialogue acts covered the cross domain grammar are:

(8)

apologize	(I'm sorry)
closing	(bye)
greeting	(hello Alex)
introduce-self	(I'm Monika)
request-repeat	(can you repeat that please)

Published in "Proceedings of AMTA-98"

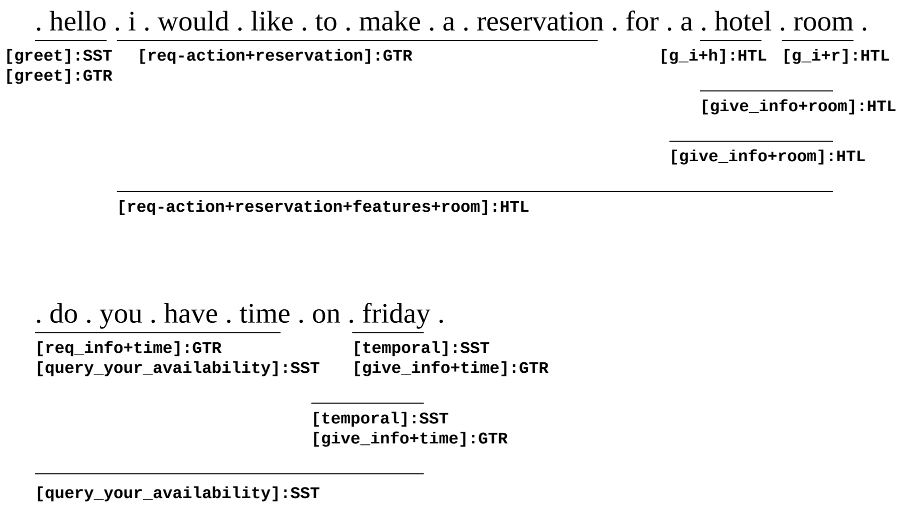

Fig. 3. DA parse lattice with multi-domain grammars.

Many of the speech acts in the cross domain grammars can be reused for new domains with only minor changes.

The shared grammars cover concepts that are used on a lower level to parse DAs in a variety of domains. Examples of concepts placed into shared grammars are date and time expressions (such as *around 5pm on Friday*) as well as lists of proper names. These grammars facilitate the expansion of the domain with new dialogue acts that are not covered by the cross-domain grammar but still contain some of the underlying concepts.

In example (6), the greeting *hello* is parsed by a cross-domain grammar, while the words *Frankfurt* and *on the fifth* are parsed by the shared grammar, in this case accessed by the sub-domain grammar for transportation.

3.3 Disambiguation with Statistical Domain Knowledge

With the expansion of our system to multiple domain grammars we have witnessed a significant increase in levels of ambiguity. It is often the case that an input utterance can consist of DAs from multiple domains. When applying the entire set of domain grammars, the utterance can be analyzed into a variety of different sequences of DAs. Disambiguating the correct sequence of DAs is crucial for correct translation.

One possible approach to disambiguation is to pre-segment the utterance in advance into sub-utterance units that are expected to belong to only one domain, and to use a domain classifier to determine the correct domain for each sub-utterance prior to or after parsing with the separate sub-domain grammars. However, this process is difficult to perform reliably, and any error in either one of the three steps would cause the translation to fail.

Published in "Proceedings of AMTA-98"

Since a great deal of knowledge about the DA segment boundaries and their domains is inherent in the modular parsing grammars, we have introduced a new approach to reduce the risk of miss-classifications: the SOUP parser parses complete utterances using *all* tagged sub-domain-grammars and produces a lattice of parse trees that contains all possible DA parses. Consequently, the parser also determines all possible ways of segmenting the utterance into DA-level segments. Figure 3 shows the resulting DA parse lattice for the utterance in example (7).

Using statistical domain knowledge, we then attempt to extract the best combination $\mathbf{S}$ of DA parse trees from the lattice. The goal is to find the most likely sequence of DAs $\mathbf{S} = s_1, s_2, \ldots s_N$ given the sequence of input words $\mathbf{W}$, i.e. to maximize the probability $P(\mathbf{S}|\mathbf{W})$. Unfortunately, the number of different dialogue acts is too large for robustly estimating their individual probabilities. Therefore, we collapse all DAs to just the domain $\mathbf{T}$ to which they belong, and use the domain tag probabilities instead of the DA probabilities. The domain tag for each DA is attached to the analysis, since it is derived from the sub-grammar from which the analysis was created. Thus, we search for the sequence of DAs $\mathbf{S} = s_1, s_2, \ldots s_N$, for which the corresponding $P(\mathbf{T}|\mathbf{W}) = P(t_1, t_2, \ldots t_N \mid \mathbf{W})$ is maximal. We perform this indirectly, using standard language modeling methods. First, applying Bayes rule, we have:

$$P(\mathbf{T}|\mathbf{W}) = \frac{P(\mathbf{W}|\mathbf{T}) \cdot P(\mathbf{T})}{P(\mathbf{W})} \tag{9}$$

Since $P(\mathbf{W})$ does not change once the utterance has been recognized, finding the maximal $P(\mathbf{T}|\mathbf{W})$ is the same as maximizing $P(\mathbf{W}|\mathbf{T}) \cdot P(\mathbf{T})$. Now, for any sequence of domains $\mathbf{T} = t_1, t_2, \ldots t_N$, we make an independence assumption between the word probabilities, such that the probability of a word w_i depends only on its domain t_i. Thus, for a given sequence of DAs $\mathbf{S} = s_1, s_2, \ldots s_N$, where the sequence of words W_i covered by the DA s_i is $W_i = w_{i1} w_{i2} \ldots w_{ik}$, we have:

$$\begin{aligned} P(W_i|\mathbf{T}) &\approx P(W_i|t_i) \qquad (10) \\ &\approx P(w_{i1}|t_i) \cdot P(w_{i2}|t_i) \ldots P(w_{ik}|t_i) \end{aligned}$$

Thus, for any possible segmentation of the entire input into a DA sequence $\mathbf{S}$, with corresponding domain sequence $\mathbf{T}$, we have:

$$\begin{aligned} P(\mathbf{W}|\mathbf{T}) &= P(W_1, W_2, \ldots W_N \mid t_1, t_2, \ldots t_N) \qquad (11) \\ &\approx P(W_1|t_1) \cdot P(W_2|t_2) \ldots P(W_N|t_N) \end{aligned}$$

where each $P(W_i|t_i)$ is calculated as in (10). To estimate each possible $P(w|t)$, the frequency of observing an individual word in the vocabulary for a given domain is estimated from a tagged training database.

The remaining needed probability for a sequence of domains $P(\mathbf{T})$ within one utterance is approximated by a unigram or a bigram statistic:

$$\begin{aligned} P(\mathbf{T}) &= P(t_1, t_2, \ldots t_N) \qquad (12) \\ &\approx P(t_1) \cdot P(t_2) \cdot \ldots \cdot P(t_N) \\ &\approx P(t_1) \cdot P(t_2|t_1) \cdot \ldots \cdot P(t_N|t_{N-1}) \end{aligned}$$

The search for the optimal DA sequence according to the probabilistic framework described above is performed within the Soup parser at the end of the parsing stage. The parser then outputs a ranked list of possible DA sequences for the entire utterance.

4 Current and Future Work

The current architecture framework of the Janus MT engine described in this paper has provided us with a solid design foundation for developing our translation system for the travel domain, which has proven to be a challenging task. Much of our current work involves incremental improvements in the coverage of our grammars and other knowledge sources and adding new languages in preparation for a thorough end-to-end evaluation. Recent preliminary end-to-end evaluations show a level of performance of about 50% acceptable translation of DAs, after about a year of system development. We aim at achieving a level of 80-90% acceptable translations within the next year.

We are also working on a number of advanced extensions to the translation system itself. These include the analysis of more advanced statistical disambiguation techniques, and the development of several alternative translation methods that we intend to combine with our grammar-based approach:

Multi-Engine Translation: Multi-engine translation was proposed by Frederking et al. [2] and has since been implemented in the Diplomat [3] and Verbmobil [11] systems. A multi-engine system applies multiple translation programs simultaneously and makes a translation by composing the best parts from the various outputs. Typically, a multi-engine system might include knowledge-based, statistical, and direct dictionary based approaches. In our case the components will be the knowledge based system described in this paper, statistical dialogue act assignment, and glossary lookups.

Combined Statistical/Grammar-based Analysis: One weakness of the grammar-based analysis system is that it is not very robust to concept phrasings that deviate significantly from those expected in the grammars, or to the occurrence of unexpected "noise" within concepts. To address this problem we are developing an alternative parsing method that combines both statistical and grammar information. Statistical information will be used in order to identify the DA, in cases where the grammar fails to do so with reasonable confidence. Using constraints from the interlingua specification, we will then predict the set of possible arguments that can our with the DA. A modified version of the grammars for parsing just argument fragments will then be used in order to extract the appropriate arguments from the utterance. Statistical identification of DAs has been investigated in the Verbmobil project [8]. Our own preliminary experiments on statistical DA extraction have shown encouraging results [4], and we are in the process of fully implementing the proposed method.

Acknowledgements

The IF formalism is the result of a close cooperation of the six C-STAR-II partners. Siemens played an important role in devising the initial format and structure. The description of the IF format is based on a specification document written by Mirella Lapata. Preliminary work on statistical DA identification was done by Toshiaki Fukada from ATR during a research term at CMU.

References

1. Thomas Bub, Wofgang Wahlster and Alex Waibel. Verbmobil: The Combination of Deep and Shallow Processing for Speech Translation. In *Proceedings of ICASSP-97*, 1997.
2. R. Frederking, S. Nirenburg, D. Farwell, S. Helmreich, E. Hovy, K. Knight, S. Beale, C. Domashnev, D. Attardo, D. Grannes, and R. Brown. Integrating Translations from Multiple Sources within the Pangloss Mark III Machine Translation', in *Proceedings of the First Conference of the Association for Machine Translation in the Americas* (AMTA-94). Columbia, Maryland, 1994.
3. R. Frederking, A. Rudnicky, and C. Hogan. Interactive Speech Translation in the Diplomat Project. In *Proceedings of the Spoken Language Translation Workshop at the 35th Meeting of the Association for Computational Linguistics (ACL-97).* Madrid, Spain. 1997.
4. Toshiaki Fukada. Statistical Extraction of Dialogue Acts for Spoken Language Translation. To appear in *Proceedings of ICSLP-98*, Sydney, Australia, 1998.
5. Marsal Gavaldà. The Soup Home Page. June 1998. `http://www.is.cs.cmu.edu/ISL.speech.parsing.soup.html`
6. Alon Lavie, Lori Levin, Puming Zhan, Maite Taboada, Donna Gates, Mirella Lapata, Cortis Clark, Matthew Broadhead, and Alex Waibel. Expanding the Domain of a Multi-lingual Speech-to-Speech Translation System. In Proceedings of the Workshop on Spoken Language Translation, ACL/EACL-97, Madrid, Spain, July 1997.
7. Laura Mayfield, Marsal Gavaldà, Y-H.Seo, Bernhard Suhm, Wayne Ward and Alex Waibel. Parsing Real Input in Janus: A Concept Based Approach. in *Proceedings of TMI-95*, 1995.
8. Norbert Reithinger and Martin Klesen. Dialogue Act Classification Using Language Models. in *Proceedings of EuroSpeech-97*, pages 2235–2238, Rhodes, Greece, 1997.
9. Alex Waibel. Interactive Translation of Conversational Speech, *Computer*, 29(7), pages 41–48.
10. Wayne Ward. 'The CMU Air Travel Information Service: Understanding spontaneous speech', In *Proceedings of the DARPA Speech and Language Workshop*, 1990.
11. Karsten Worm. A Model for Robust Processing of Spontaneous Speech by Integrating Viable Fragments. In *Proceedings of COLING-ACL-98*, Montreal, Canada, August 1998.