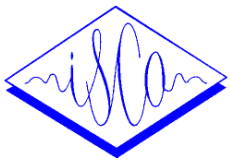

4th European Conference on
Speech Communication and Technology
EUROSPEECH '95
Madrid, Spain, September 18-21, 1995

Concept-Based Spontaneous Speech Understanding System

Esther Levin Roberto Pieraccini

Speech Research Department
AT&T Bell Laboratories
600 Mountain Avenue
Murray Hill, NJ 07974, USA

Abstract

In this paper we describe the issues in the design of CHRONUS - a spontaneous speech understanding system. The CHRONUS system is based on the approach we proposed in 1991, which formalizes the understanding problem as a communication problem. The model assumes that a spoken sentence is generated by a HMM like process whose hidden states correspond to elemental meaning units called *concepts*. Understanding therefore consists in decoding the hidden concepts given a spoken utterance [2]. The CHRONUS system was used in the ARPA ATIS task, a spontaneous speech database interface. In the 1994 official evaluation it achieved a state of the art accuracy. In the following paper we discuss the features of the system that contributed to this success.

1 Introduction

There are currently two main paradigms driving the development of speech understanding systems: the corpus based and the linguistic. In the corpus based paradigm the understanding system is parametrized and its parameters are learned from an annotated corpus. In the systems based on linguistic approach the necessary linguistic/syntactic/semantic knowledge is hand-coded into the system usually in the form of rules. CHRONUS is a hybrid system: its core is a stochastic model whose parameters are learned from a corpus, but all the other components are hand-coded. The hybrid approach combines the advantages of both corpus based and linguistic paradigms and is based on two principles:

- *Everything that can be learned from available data should.* A corpus is an invaluable source of knowledge that can be used for training the models used at different stages of processing. This is especially important when dealing with spontaneous speech where grammars obtained just by introspection cannot account for phenomena that can be encountered in a corpus of transcribed speech. The conceptual model in CHRONUS is learned from annotated sentence transcriptions.

- *Rather than attempting to learn complex and rare linguistic events, provide convenient ways to incorporate the established linguistic knowledge into the system.* This is achieved by proper representation of the knowledge. The knowledge representation in CHRONUS is based on the following principles:

  - *Separation among algorithms, general and task specific knowledge.* This makes the system suitable for incremental improvement. For many system that show high performance in standard tasks is impossible to distinguish between the knowledge and the mechanism that uses it, since they are rather an intricate combination of `if ... then ...` statements. Although the original designers of the system may be able of changing or upgrading it, this is practically impossible or very difficult for anybody else.

    Patchability. The designer of a system should be able to introduce knowledge that either cannot be learned from data or is already available. For instance, if at a certain point of the assessment of the system the designer finds out the the expression *no later than*, that was never observed in the training corpus could be used as a synonym of *before than*, he should be able to upgrade the system in order to account for that.

2 Understanding and Meaning Representation

As shown in Fig.1 our understanding system is composed of three main modules. First, the speech recognizer transcribes the spoken input sentence [4]. Then the meaning interpretation module converts this transcription into a formal representation of the meaning conveyed by the sentence in the context of previous sentences. The backend converts the meaning representation into the desired action. The meaning representaion is task dependent, it covers the semantic domain of the task, and is unambiguously interpreted by the backend. In CHRONUS,

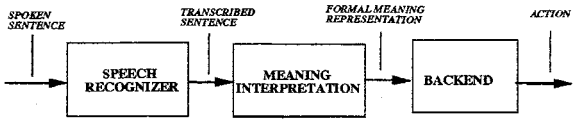

Figure 1: A Speech Understanding System

```
ORIGIN_CITY: NNYC
DESTINATION_CITY: SSFO
WEEKDAY: SATURDAY
TIME: 0<1200
AIRCRAFT: 74M
```

Table 1: Example of template

the output action consists in retrieval of data from a relational database, and the backend is a relational database interface. The meaning of the request is represented in the form of a template, i.e., a set of tokens, where each token is a keyword/value pair. For example, a sentence like

> *I WOULD LIKE TO GO FROM NEW YORK TO SAN FRANCISCO SATURDAY MORNING I PREFER TO FLY ON A BOEING SEVEN FORTY SEVEN.*

is represented by a template like in Table 1. The keywords are related to the attributes of the database, and the values are represented in the same format as are the entries of the database. The relational database interface takes the meaning representation in the form of a template and extracts the information that was requested from the database. The conventional way of doing this consists in writing a set of transcription rules that transforms the template into an SQL statement. The complexity of this approach is high, and the resulting code is hard to maintain and to port to other applications because it depends on the structure of the database. Our approach is based on the principle of separation between algorithms and knowledge that allows the use of the same module for different tasks with minimal human effort. The task specific knowledge (i.e. the particular database used) is represented by a network that can be easily derived from the database schema. The network is navigated through a *constraint propagation algorithm* that extracts the proper tuples.

The meaning interpretation module in our system is further divided into two main components, corresponding to local and global analyses, as shown in Fig. 2. This architecture reflects one of the main assumptions we make about the nature of spontaneous speech. We assume that a spoken sentence can be segmented into phrases, where each phrase corresponds to one unit of meaning, i.e., a concept, that is described in our meaning representation as a keyword/value pair. This locality assumption enhances the robustness of the system against unexpected *non linguistic* phenomena and speech recognizer mistakes. This means that the system may not be able to deal with the complexity present in sentences like:

> *Show me the departure time of the flight from Boston to Denver that arrives in Denver at least forty minutes before the cheapest Delta flight from Denver to San Francisco.*

but allows sentences like:

> *I want a flight to — no excuse me — from Dallas — uhm — I am going to Boston — uhm — do you have anything with Delta?*

The last kind of complexity, related to false starts, abrupt changes of subject, ungrammaticalities, noise, etc., is the one that characterizes spontaneous speech. Moreover, in an application with a relatively simple domain, one can exclude, as a first approximation, the use of complex recursive syntactic forms and logically complex questions. Therefore the first component of Fig. 2 performs a local analysis of a sentence, segmenting a sentence into phrases, and outputing a token describing each phrase. The set of these tokens constitutes the

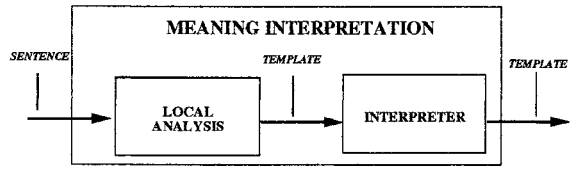

Figure 2: The Meaning Interpretation Module

output template of this module, as in the previous example described in Table 1. The analysis of a sentence as a whole is delayed to the second module, the interpreter, that resolves remaining ambiguities by exploiting non-local relationships between tokens. This module is also responsible for interpreting each new sentence in the context of previously spoken sentences. It outputs the final template that is then processed by the backend. For instance `TIME` in the previous example could be either `ORIGIN_TIME` or `DESTINATION_TIME`, the `SUBJECT` of the query is missing, and the information derived from previous sentences is not taken into account. For instance, if the previous sentence was *I generally fly with United Airlines*, the result of the interpreter is a template like the one shown in Table 2. The interpreter is based on a set of hand-coded rules. Development tools and convenient knowledge representation in the form of semantic network facilitate the design of this module.

3 Stochastic Approach to Understanding

The local analysis module of Fig. 2 is based on a conceptual stochastic model whose parameters are estimated

```
AIRLINE: UA
ORIGIN_CITY: NNYC
DESTINATION_CITY: SSFO
WEEKDAY: SATURDAY
ORIGIN_TIME: 0<1200
AIRCRAFT: 74M
SUBJECT: FLIGHT
```

Table 2: Example of template generated by the interpreter

```
wish: I WOULD LIKE TO GO
origin: FROM NEW YORK
destin: TO SAN FRANCISCO
day: SATURDAY
time: MORNING
aircraft: I PREFER TO FLY ON A BOEING SEVEN FORTY SEVEN
```

Table 3: Example of conceptual segmentation generated by the conceptual decoder

from the corpus. According to this model the sequence that represents the spoken sentence (either transcribed words, or the acoustic observations) is considered to be the output of a noisy channel whose input is the meaning represented by a template. Therefore the problem of understanding is reduced to that of maximum a posteriori probability decoding of the template given the sentence. Fig. 3 shows a more detailed view of the local analysis module. The second module, the conceptor, imple-

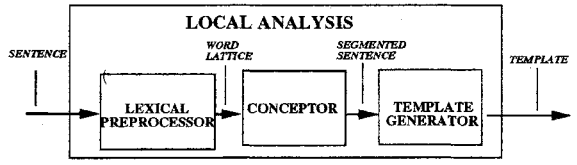

Figure 3: The Local Analysis Module

ments the maximum a posteriori probability decoding. It is based on an HMM [2] whose states correspond to the different concepts and are characterized by stochastic language models describing the phrases that express the corresponding concept. The decoding consists of a Viterbi search producing the most likely segmentation of the incoming sentence into phrases labeled by concepts . For our running example the segmentation is shown in Table 3. The template generator analyses locally each labeled phrase, producing the corresponding token value, and generating the template shown back in Table 1.

The function of the lexical preprocessor is to improve the estimation of the stochastic model by introducing word classes i.e. airports, airlines, numbers, etc. The output of the lexical preprocessor is a lattice of words, where each entry consists of a word hypothesis and its beginning and ending indices. Given the limited semantic context the semantic association of most words is unique,

```
 0  1 I                    14 15 PREFER(S)
 1  2 WOULD                15 16 TO
 2  3 LIKE(S)              16 17 FL(Y|IES)
 3  4 TO                   17 18 ON
 4  5 GO(ES)               18 20 [A](<aircraft_make>BOEING)
 5  6 FROM                 18 23 [A](<aircraft>74M)
 6  7 NEW                  19 20 (<aircraft_make>BOEING)
 6  8 (<city>NNYC)         19 23 (<aircraft>74M)
 6  8 (<state>NY)          20 21 (<numbers>7)
 8  9 TO                   20 22 (<numbers>740)
 9 10 SAN                  20 23 (<numbers>747)
 9 11 (<city>SSFO)         21 22 (<numbers>40)
11 12 (<day_name>SATURDAY) 21 23 (<numbers>47)
12 13 MORNING(S)           22 23 (<numbers>7)
13 14 I
```

Table 4: Example of lattice generated by the lexical analyzer

for instance *SAN FRANCISCO* is unambiguously the name of a city and *SATURDAY* is unambiguously a day name. However, some ambiguities can still remain (e.g. *SEVEN FORTY SEVEN* could be the identifier of an aircraft as well as the number of a flight or a time specification, moreover it can be parsed in several ways: 747, 7-40-7, 7-47). Since at this level of processing it cannot be decided yet which the correct semantic interpretation is, the lexical analyzer generates multiple hypotheses arranging them in a lattice structure, like the one in Table 4.

4 Experimental Evaluation

The application of the understanding system reported in this paper refers to the ARPA ATIS (Air Travel Information System) task [1]. The ATIS task is built around a relational database, a subset of the Official Airline Guide. The corpus consists of about 20,000 spoken utterances collected through a *wizard* system and carefully annotated and transcribed. The 1994 evaluation set consisted of 1000 independently collected sentences.The assessment of the participating systems is performed by comparing the output of the system with correct answers determined by the annotators. Fig. 4 summarizes the results of this evaluation [3]. Three standard tests were performed. The speech recognition test measures the accuracy of the speech recognizer as the overall percentage of correct words, the natural language test measures the overall percentage of correctly answered sentences (i.e. those for which the given answer matched the reference answer) of the natural language component alone (i.e. the system input was the textual transcription of each answerable sentence), and spoken language understanding test measures the overall percentage of correctly answered sentences for the complete systems (i.e. the system input was the actual speech). The performance of our system in the natural language test is the highest compared to the other systems. However more impor-

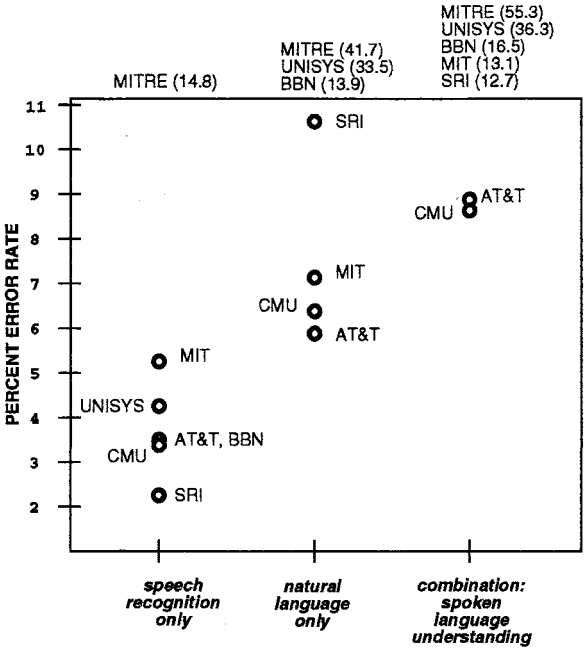

Figure 4: Results of 1994 ARPA ATIS evaluation.

tant than the actual performance on the ATIS task is the flexible architecture we developed that not only allowed us to build a high performance system in a very short time, but possibly allows for a rapid development of other applications.

5 The CHRONUS System: a Summary

The functional diagram of the system we implemented according to the principles we have stated above is shown in Fig. 5. The characteristics of the system are the following:

- The parameters of the conceptor are estimated for a corpus of annotated sentences.
- All the modules, except the interpreter, are implemented through table driven general programs, meaning that each one of them can be customized to other applications by changing the associated data files.
- All the modules, except the interpreter, are local, in the sense that they work on limited portions of the input sentence.
- All the necessary non local processing is concentrated in the interpreter.

This architecture represented for us the attainment of two most important goals: the achievement of the highest accuracy in the natural language test of the 1994 ARPA ATIS [3] evaluation, and the development of a first prototype of a general toolkit for the design of speech understanding systems in the same application class.

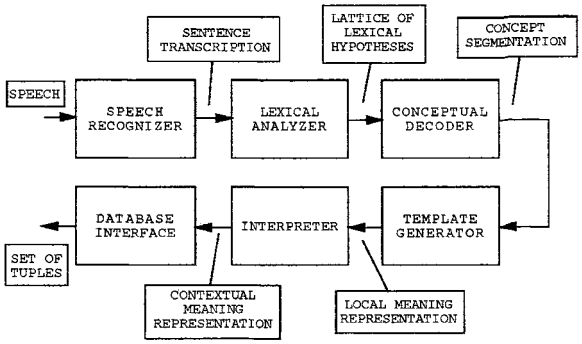

Figure 5: A functional diagram of CHRONUS, the AT&T speech understanding system

References

[1] *MADCOW*, "Multi-Site Data Collection for a Spoken Language Corpus,", *Proc. of Fifth Darpa Workshop on Speech and Natural Language*, Harriman, NY, Feb 1992.

[2] Pieraccini, R., Levin, E., "Stochastic Representation of Semantic Structure for Speech Understanding," *Speech Communication*, Vol.11 pp. 283-288, 1992.

[3] Levin, E., Pieraccini, R. "CHRONUS, The Next Generation," Proc. of 1995 ARPA Spoken Language Systems Technology Workshop, Austin Texas, Jan. 1995.

[4] Bocchieri, E.L., Riccardi, G. Anantharaman, J., "The 1994 AT&T ATIS CHRONUS Recognizer," Proc. of 1995 ARPA Spoken Language Systems Technology Workshop, Austin Texas, Jan. 1995.