ISCA Archive
http://www.isca-speech.org/archive

3rd European Conference on
Speech Communication and Technology
EUROSPEECH'93
Berlin, Germany, September 19-23, 1993

INTEGRATION OF NEURAL NETWORKS AND ROBUST PARSERS IN NATURAL LANGUAGE UNDERSTANDING

Ying Cheng†‡, Yves Normandin† and Paul Fortier‡

†CRIM, Suite 800, 1801 McGill College Ave.
Montréal, Québec, Canada, H3A 2N4
‡ Dept. of Electrical Engineering, Laval University
Québec, Québec, Canada, G1K 7P4

10.21437/Eurospeech.1993-306

ABSTRACT

In this paper, we describe a natural language understanding system which focuses on spoken language and integrates a neural network classifier and robust parsers. With the help of classification, the application domain is divided into several subsets. Parsers are constructed for each subset and as a result, the complexity of grammar construction is much smaller than if a grammar for the whole application domain had to be constructed. Furthermore, by using a neural network, our goal was to take the advantage of its learning ability and robustness to noise.

Although the system was implemented in a short time, it performed reasonably well in its first participation in a DARPA ATIS (Air Travel Information System) evaluation in Nov. 92. It was quite robust to speech recognition errors in particular.[1]

Keywords: *neural network, parser, natural language understanding.*

1. INTRODUCTION

Spoken language understanding systems are different from those specifically built for written text in that they have to anticipate grammatical errors and, possibly, speech recognition errors. Strategies merely based on rigid grammar formalisms will not likely be robust to such phenomena. In this paper, we describe a natural language understanding system which attempts to improve robustness and generalization capabilities by relying on a combination of local parsers, a neural network classifier and class specific context-free grammar based parsers.

The paper is organized as follows. Section 2 introduces the ATIS domain to which the system is applied. In section 3, we describe the natural language understanding system in general. In section 4, the implementation of the neural network classifier and robust parsers are discussed. The results of the evaluation and discussions are given in section 5. Section 6 concludes the paper.

[1]This research was conducted in collaboration with Alex Informatique, CAE Electronics, CRIQ, IREQ, and Tektrend, as part of a research effort partly sponsored by the Canadian Ministry of Industry, Science and Technology (ISTC).

2. ATIS DOMAIN

ATIS is a spoken language understanding task developed and sponsored by DARPA. It had its first benchmark evaluation in June 1990 and its fourth one took place in November 1992.

ATIS uses spontaneous speech in a normal office setting from participants engaged in a travel planning task. The participants are asked to carry out the given travel plans by interrogating the air travel information system. The provided informations are extracted from the Official Airline Guide (OAG). There are five sites involved in ATIS data collection [3], and the procedure varies from site to site. Thus, it leads the great diversities in the ATIS data.

In an ATIS evaluation, there are two tests related to natural language understanding. In the first test, the input to the natural language part is a transcript sentence. This test is named the NL test. In the second test, called SLS, the natural language understanding part is integrated with the speech recognition system so that its input is the output of the speech recognition system. The sentences are classified into three different categories as follows:

- **A,** the sentences which can be interpreted without any help from dialogue history.
- **D,** the sentences that should be interpreted with the help from dialogue history.
- **X,** the sentences which are unanswerable due to several reasons.

3. OVERVIEW OF THE NATURAL LANGUAGE UNDERSTANDING SYSTEM

The typical configuration of an ATIS spoken language system is illustrated in Figure 1. The goal of the NL part is to convert the input word sequences (i.e., sentences) into a frame of intermediate representation to be used by the SQL module. This module accesses the database in order to retrieve the requested information and display it to the user. Each intermediate representation frame consists of two parts: 1) the requested attributes that are provided to the users; 2) the constraints.

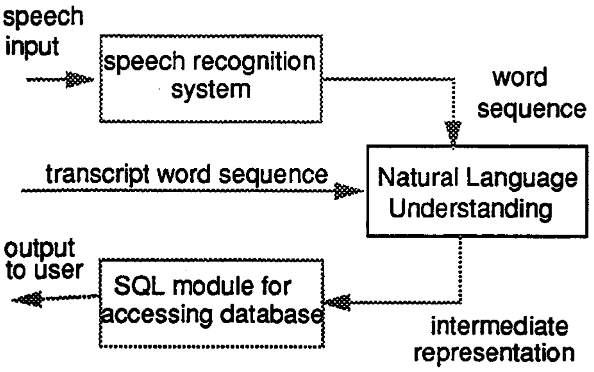

Figure 1, the spoken language system for ATIS task

3.1. Architecture

Figure 2 gives the natural language system's architecture with an example sentence. The main system components are a neural network classifier and three different types of parsers.

The reason for using a classifier is to reduce the complexity of grammar construction for the parsers which will be discussed later. The categories are defined according to the table names corresponding to the requested attributes of the sentence [1] [2].

In spite of typical spoken language phenomena such as loose structure, repetition, insertion and ungrammatical errors, the meaning of the sentence usually remains clear. This is due to the relatively well structured semantic fragments of the sentence. Based on this observation, we use a segment and tag parser which acts locally and parses semantic fragments in the sentence. Then, the requested attribute parser and case-frame based parser work on the segmented and tagged sentence in such a way that possible errors caused by the above mentioned phenomena can be avoided. More details will be discussed in section 4.

With the help of the example sentence given in Figure 2, we explain the functions of each module as follows.

Preprocessor: Turns the separate letters into abbreviations, converts numbers into arabic numbers, and so on. For instance, the three letters "*T W A*" in the example sentence, are turned into the abbreviation "*TWA*".

Neural network classifier: Predicts the categories to which the sentence belongs. The result of classification is a prediction result and its score. For example, {5, 0.957694} indicates category 5 with a score of 0.957694, where category 5 corresponds to sentences whose requested attributes are those about the flight table. More details will be discussed in section 4.

Segment and tag parser: In parallel with the classification, the segment and tag parser is used to parse the semantic fragments and tag them. For instance "*FROM DALLAS*" is segmented as a semantic fragment with the tag "location_1".

Requested attribute parser: Attempts to parse the requested attributes of the sentence. For the given sentence in Figure 2, the output is "flight.flight_id"

Segment interpreter: This interpreter is used to map the semantic fragments into pragmatic representations. For instance, the fragment "*FROM DALLAS*" in the example sentence is mapped into "flight.from_airport DDFW".

Case frame based parser: Its purpose is to put together the requested attributes and constraints. The constraints are those semantic fragments found by the segment and tag parser. Scanning from left to right, the parser analyses each segment and passes it along with its relation name to the segment interpreter which returns with the pragmatic representation. Then the parser fills it into the constraint sub-frame.

Context manager: This is used to analyse the dialogue history and provide history information to the case frame based parser that will further modify the interpretation if necessary.

Generator of intermediate representation to SQL: Using the output from the case frame based parser, an the intermediate semantic representation for the SQL module is generated so the SQL module can produce the correct database query.

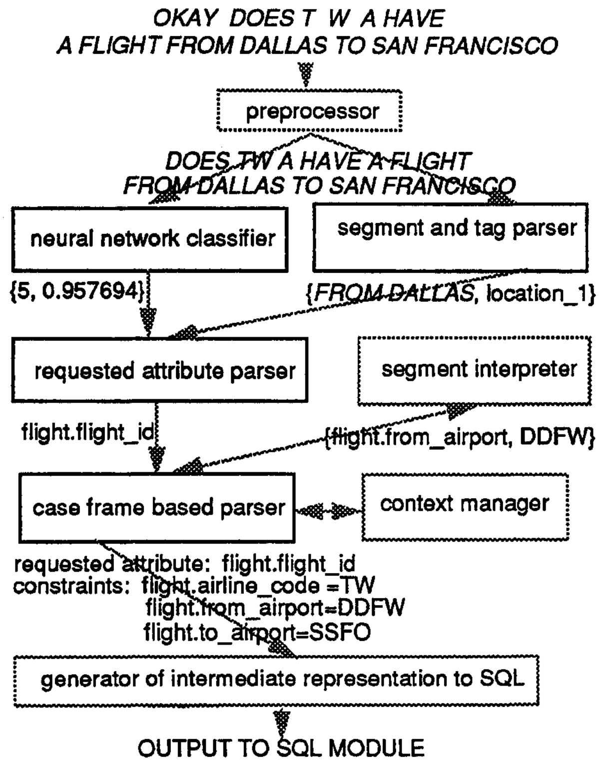

Figure 2

4. IMPLEMENTATION

4.1. Neural network classifier

A three layer perceptron is used to classify sentences. As mentioned above, the goal of classification is to reduce the complexity of grammar construction by dividing the problem into several sub-problems. For the ATIS application, there are ten sentence categories [1]. The classes correspond to the table names in the OAG database.

Representation

In order to apply the neural network to the sentence classification problem, sentences have to be encoded in such a way that they can be used as input to the network. Also, the output units have to be defined. We empirically defined the following representations:

- *word representation:* 55 semantic categories are manually chosen to represent a word. Each category has one bit.
- *input representation:* A vector of the keywords in the sentence is presented to the neural network. The position of the keyword in the vector corresponds to its order in the sentence. The length of the vector is limited to fifteen.
- *output representation:* One output unit represents one category.

Architecture

Figure 3 gives the architecture of the neural network.

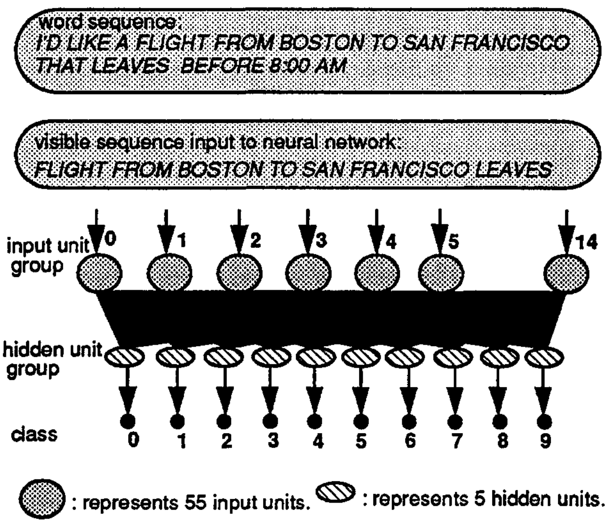

Figure 3, neural network classifier

Connections: the input layer is fully connected with the hidden layer while the hidden layer is partly connected with the output layer such that each output unit has its own five hidden units.

Units: 55×15 input, 5×10 hidden and 10 output units.

Training

The backpropagation [4] algorithm was used in training. The training data were derived from the annotated ATIS training data and were automatically labelled from the SQL queries which were provided with each sentence by NIST. Different data were used for the NL and SLS tests. 1514 class-A transcript sentences were used for training for the NL test. The classification rate on the training data was 96.2%. 1527 sentence from CRIM's speech recognizer were used to train for the SLS test. Its classification rate was 96.0%.

4.2. Robust parsers

Segment and tag parser

The segment parser is mostly based on semantic knowledge. It uses a context-free grammar and acts locally. It is designed to tolerate repetitions and insertions of unimportant words. There are fourteen different semantic fragments such as "location", "time" and so on. All the semantic fragments are related to constraints. The rules of the parser are abstracted from the ATIS training data.

Requested attribute parser

Ten parsers are used to extract the requested attributes from the input sentence. The parsers use a context-free grammar and are based on both semantic and syntactic knowledge. Each parser corresponds to a category, that is, the grammar of the parser only applies to its corresponding category. A parser's grammar is constructed by studying the annotated training data which fall into its category. This "divide and conquer" strategy allowed us to implement the parsers in a short time because the complexity of grammar construction on a subset of the application is much reduced if compared to that on the entire application. Moreover, computation efficiency is improved due to the relative small parsing trees.

Only the parsers whose categories are predicted by a neural network classifier are active. The input is the segmented and tagged sentence. Therefore, the parsers work on the sequence which contains the semantic fragments and the remaining un-segmented words. The parsers only attempt to find the requested attributes mentioned in the sentence and not to analyse the structure of the entire sentence, and therefore ignore the possible errors occurring at some unimportant positions of the sentence. The output of a parser are the pragmatic representations of the requested attributes. For example, for the query "*WHAT ARE THE FLIGHTS FROM DENVER TO BOSTON*", the pragmatic representation is "flight.fight_id".

Case frame based parser

The case frame based parser builds the interpretation frame which consists of a requested attribute sub-frame and a constraint sub-frame. The output of the requested attribute parser is directly taken to fill the requested attribute part.

Scanning from left to right, the parser analyses the encountered semantic fragment and passes the fragment and a table name to the segment interpreter which returns the pragmatic representation of the fragment. The table name for a semantic fragment is defined by a case-frame. A case-frame is derived from the structure of the relational database and defines the possible table names for a certain semantic fragment when a requested attribute is known.

5. EVALUATION RESULTS

In combination with CRIM's speech recognition system[1], the natural language understanding system was evaluated in the DARPA ATIS Nov'92 benchmark test. Because of the short time available for implementing the whole spoken language system, our SQL module was not completed. Some of the situations such as queries related to flight.flight_leg and multi-relation queries could not be properly treated at that time.

NL test

In this test, the input to the system are the transcript sentences. The official results are given in table 1 which presents the results of class A and D together, class A alone and class D alone.

TABLE 1. RESULTS OF NOV92 ATIS BENCHMARK TEST (NL part)

class	#T	#F	#NA	#Utt	WEIGHTED ERROR
A+D	497	158	19	674	49.7
A	356	62	9	427	31.1
D	141	96	10	247	81.8

Among the errors, around 20% arose from simple errors in the program and misunderstanding of the given interpretation principles. Only around 20% of the errors are somewhat serious, most of which being related to the grouping problem.

Also, because of the uncompleted context-manager, class D had much more errors than class A. Around 25% of the errors resulted from this.

The remaining errors were the result of an unfinished SQL module, an incomplete coverage of the parsers' grammars, and not well-tuned classifier errors. But these errors can be fixed without many difficulties.

The neural network had around 83% classification rate on class A sentences and around 76% classification rate on class D. In class D, one typical misclassification situation happened when the sentence contained ellipsis or anaphora. The following is an example:

The first sentence: "*SHOW ME THE FLIGHT FROM DENVER TO BOSTON*"

The second sentence: "*THE EARLIEST ONE PLEASE*".

Since, the neural network does not remember the history, it fails to classify the second sentence correctly. This is the shortcoming of the current neural network version. Such shortcoming contributed as well to the false interpretations.

SLS test

In the SLS test, the natural language understanding part was loosely incorporated with CRIM's speech recognition system. The input of the natural language understanding system is the one best hypotheses of the speech recognition system. The CRIM's speech recognition system got 88.7% word accuracy and 56.4% sentence errors on class A and D.

TABLE 2. RESULTS OF NOV92 ATIS BENCHMARK TEST (SLS part)

class	#T	#F	#NA	#Utt	WEIGHTED ERROR
A+D	398	231	45	674	75.2
A	296	113	18	427	57.1
D	102	118	27	247	106.5

Most of the errors found in the SLS part but not in the NL part were due to misrecognition of keywords, such as numbers and names of months. Apart from this, the system performed quite robustly to the recognition errors.

In this part of the test, the classification rates on class A and class D were, respectively, around 80% and 74%.

Discussion

By analysing the evaluation results, we found that there is still room for improving the performance. Steps such as completing the context manager, fixing simple bugs, modifying the neural network classifier to handle the context dependent cases and so on, should help substantially

6. CONCLUSION

We discussed a natural language understanding system which integrates a neural network classifier and robust parsers and focuses on generalization over the phenomena in spoken language. From the reasonable good results in its first participation, it can be concluded that the strategies used in this system are quiet promising.

REFERENCE

[1] *Cardin R., et al.:* "CRIM's Speech Understanding System for the ATIS Task", notebook of DARPA's spoken language technology workshop, MIT, Jan., 1993

[2] *Cheng Y., Normandin Y., and Fortier P.:* Topic Identification Using a Neural Network with a Keyword-Spotting Preprocessor, Proceedings of international conference on spoken language processing, Banff, Canada, Oct. 1992.

[3] *MADCOW*: Multi-Site Data Collection for a Spoken Language Corpus. Proceedings of the Fifth DARPA workshop on speech and natural language, Harriman, NY, Feb., 1992.

[4] *Rumelhart D. E., Hinton, G. E. and Williams, R.J.:* Learning Internal Presentations by Error Propagation, in Parallel Distributed Processing, the MIT press, 1986.