ADAPTING PROBABILITY-TRANSITIONS IN DP MATCHING PROCESS FOR AN ORAL TASK-ORIENTED DIALOGUE

K. MATROUF, J.L. GAUVAIN, F. NEEL, J. MARIANI

LIMSI/CNRS B.P.133 91403 ORSAY cedex FRANCE

Abstract

In developing an oral dialogue system for air-traffic controller training, we had two main objectives in mind: we aim to make it both user-friendly and robust. Different knowledge sources including the vocabulary, the phraseology, the task model and the history of dialogue, are evolving during the dialogue. These knowledge bases are represented in a unified form of hierarchical frames. The paper focuses on the interaction between the recognizer and the dialogue manager, or, more precisely, on the representation of the language model used to limit the DTW search space and on how it is modified by the dialogue manager in order to permit or prohibit messages and improve recognition. In system evaluation, the classical DTW approach is compared to 2 probabilistic approaches: without and with dynamically updated transition probabilities.

1 Introduction

The system presented here has been developed to evaluate the introduction of voice technologies in air-traffic controller training. The study has been carried out in collaboration with the CENA (National Research Center for Air Traffic Control). The dialogue system was designed to replace the "pseudo-pilot", a human playing the role of the pilot during training exercises, and communicating, both with the student controller and with the air-traffic simulator which updates the radar image on a screen. The language used, an official "phraseology", belongs to an "operative type" [3]. The dialogue concerns a restricted semantic domain, uses limited syntax and vocabulary and aims at the execution of a precise task. Requests coming from the controller are interpreted, expressed in a formal command language and sent to the simulator.

In this paper, the knowledge representation which renders the cooperation of different knowledge sources possible is described. Our approach makes use of pragmatic knowledge to predict a sub-language which dynamically limits the recognition search space and therefore improves its accuracy.

2 Knowledge representation

The model proposed here is based on frame theory [2]. A frame is a data structure which describes each object or concept with a number of named slots which hold data values or properties as well as procedures. The knowledge representation was chosen since information coming from different levels can be represented with the same formalism.

Messages used in the air-traffic control (ATC) language can be classified in several categories (for example: heading, level, speed). A frame is associated to each message category to organize a set of frames called concepts. All knowledge required for message analysis and interpretation is present in the frame of categorie. A message consists of a call-sign and a question, an instruction or an information. An example showing the data organization in a frame corresponding to the *"level"* category is given in Fig 1 using the Lisp representation.

Frames are organized in a hierarchized unified structure, and constitute a part of the static knowledge bases as shown in Fig 2, together with the word confusion matrix which is used for acoustic correction [1].

For each plane present in the air sector, the system possesses an image of the plane containing all the information concerning it. These parameters are constantly updated during the dialogue and constitute the model of the task which we call the *"context"*. The system also keeps a trace of the preceding exchanges, in the dialogue history. This information constitutes the dynamic knowledge bases.

```
(*level
 (ISA          *instruction)
 (call-sign    LINK *ind)
 (action       VAL (maintain descend ...)
 (subject      VAL ( level )
 (param        LINK plevel )
 (constrain    (param.value mul 5)
               (param.value <= 600)
               (param.value >= 90))
(plevel
     (digit1 VAL (@digit))
     (digit2 VAL (@digit))
     (digit3 VAL (@digit))
     (valeur FUNC calcul))
```

Fig. 1 Example of a frame (''level'' category

CH2847-2/90/0000-0569 $1.00 © 1990 IEEE

3 Dialogue module

The dialogue between an air-controller and a pilot respects rules which are called a "phraseology". Whenever an air-controller asks a question about a parameter, the pilot must answer the value of this parameter. When the air-controller gives an instruction, the pilot must repeat the instruction to demonstrate that he understood it.

The dialogue module is the central module of the system: it coordinates the other modules involved, the contextual interpretation of the messages coming from the task (simulator) or the user (controller). The structure the system has at its disposal is a network of frames, called the **dialogue network**, which represents the dialogue current state.

Message analysis consists of determining the message category and then instantiating the corresponding frame.

Instantiation consists of collecting information from the message and filling the frame slots.

The frames provided by the analyzer are merged in the dialogue network. After analysis, the remaining modules are invoked in the following order:

1 - error detection and correction,

2 - message generation to the simulator and the user,

3 - updating of the dialogue network and of the knowledge bases,

4 - prediction for the next message.

More details about error detection and correction are given in [1].

4 Prediction

Predictions essentially consist of using understanding upper levels of knowledge (syntactic, semantic and pragmatic) in order to limit the recognizer search space for words and to improve the recognition performance [5]. The predictive Several types of predictions can be distinguished: 1) syntactic predictions, 2) semantic predictions [6], 3) pragmatic predictions.

Speech recognition systems make frequent use of syntactic and semantic predictions but more rarely use pragmatic

Pragmatic predictions consist of using of the correspondence between two successive utterances taking into account the task allows to considerably limit the recognition search space, especially for a structured task-oriented dialogue. For such an approach to be efficient, a complete model of task universe is required. We propose a method aimed at predictively utilizing this knowledge with direct effect on syntax rules.

Pragmatic predictions may be of 2 types in our system:

1) prediction or exclusion of a grammar subset,

2) reinforcement or diminution of occurrence probabilities of one subset, without cancelling probabilities of other message types.

4.1 Definition of pragmatic predictions

Predictions are dependent not only on the last message, but more generally on the dialogue history and on the current context.

There are some examples of predictions associated with the controller/pilot dialogue:

1) **punctual exchange**: when the system asks a question to the student-controller, the universe of response is limited and the system can predict the response category.

2) **dialogue history**: before the controller gives the pilot an instruction concerning a parameter, he generally asks for the present values of the parameter concerned.

Predictions are described by rules in which all the knowledge sources cooperate. Rules are activated before each recognition process. Each rule is defined in the following way:

Rule ::=< *condition* >   < *prediction* >

< *prediction* >::= $(C_0, k_0)...(C_n, k_n)$

In which the probability of occurrence of a concept C_i, is augmented by k_i, where k_i is a weight. A weight is a multiplier coefficient. The weight k_i expresses the degree of the prediction. Ideally, the weights should be determined by counting the occurrences of each precondition and conclusion in the corpus for each rule.

However, since our corpus was incomplete (we did not have the pilots text) we could not accurately compute these values, therefore we were forced to estimate them.

The estimates were chosen based on discussions with researchers from CENA and by listening to short excepts of pilot/controller dialogs.

Number of rules	210
Vocabulary size	210
Language size	517.10^{16}
Dynamic branching factor	24.2
Sentence length	12

Table 1: Grammar characteristics

4.2 Dialogue and syntax handling

Syntax is the only means that the dialogue system has to constrain recognition; therefore all the predictions must be incorporated in the syntax used by the recognition system. However dialogue requires a dynamic syntax according to the current context. Different solutions may be envisaged:

We use a probabilistic syntax for recognition and the system modifies the rule probabilities according to the context. This solution was chosen, because it allows to take into account the probabilistic prediction. The language is described by a binary grammar. A binary grammar is generated from a context-free grammar and the probabilities from a corpus.

4.3 Syntax construction

The syntax is constructed taking into account the official phraseology and thei syntax really used in operational conditions. The syntax arcs are augmented by probabilities reflecting the frequency of occurrence.

A corpus was obtained in operational conditions at the CENA: it contains 10,000 words taken from a vocabulary of 210 words. Grammar characteristics are given in table 1.

4.4 Prediction propagation in the syntax network

Each concept is associated with a set of words used to specify it. For example, the concept **"heading"** can be associated to the messages *"take heading 200"* or *"turn right heading 300"*. The list of the words which corresponds to the concept *"heading"* includes : *"maintain, continue, take, turn, left, right, heading, degrees ..."*

Knowing the words associated with a concept, the system is able to find the arcs of the automaton corresponding to the concept. This is straight-forward because the binary syntax representation associates only one arc to one word.

The initial grammar has probabilities obtained from training. A concept prediction is propagated by changing the arc probabilities.

One word can be associated with several concepts. Therefore when a concept is eliminated, one can not simply rule out all of the words associated with that concept. Instead, the word probabilities are reduced by a factor dependent upon the importance of the relationship between the word and the concept.

For example the word *"maintain"* can be associated with the concepts *"heading"*, *"level"*, ...; we can say *"maintain your heading"* or *"maintain level"*.

Each word (arc) is therefore associated with a number of related concepts.

The new probability of the word w is:

$$p_w = p_w^0 \times (\frac{n_w - l + \sum_{i \in C(w)} k_i}{n_w})$$

p_0: initial probability for word w,
n_w: total number of concepts associated to the word w,
$C(w)$: set of concepts to be modified,
l: number of elements in set $C(w)$,
$k_1 ... k_i$ the corresponding weights.

These new probabilities are normalized such that the sum of the arc probabilities leaving a node is equal to 1.

4.5 Use of probability by the recognition system

The recognition system used is AMADEUS designed at LIMSI and commercialized by VECSYS. It is a speaker-dependent, continuous speech system, using a DTW algorithm [4]. It can process a vocabulary size on the order of 200 to 300 words in real-time, due to the use of a DTW specialized chip also designed at LIMSI [7]. AMADEUS uses a regular grammar represented in the form of a table, comprising all the arcs entering and leaving the node. Probabilities can be associated with the forward arcs. The system was adapted in order to use transition probabilities in the recognition process.

The recognition process consists of finding the sequence of words which minimizes the accumulated distance D and maximizes the probability of occurrence P. This is equal to minimizing: $(D - a * log(P))$. For each word transition, the retained local distance is therefore:

$$distance = d - a * log(p_i)$$

a : conversion constant,
d : distance obtained by DTW,
p_i: transition probability.

5 System evaluation

The objective of the tests was to evaluate the importance of upper-level understanding in improving recognizer performance. The upper level knowledge was used in two different ways. The first directly intervened in the recognition process enhancing possible solutions. The second made use of internal correction strategies for semantic recognition.

6 speakers (5 male, 1 female) took part in the tests. For the testing phase, 5 scripts were defined each with an average of 20 sentences (4,930 words). Each speaker was placed in a realistic simulation condition, watching the planes moving in the sector on the screen, pronouncing the written instructions or questions and answering the system initiatives.

Test results are presented in Table 2. The acoustic recognition results are the output of the recognizer. Semantic recognition means that the message was correctly understood by the system, despite recognition errors. Results are given using a classical DTW algorithm, with word transition probabilities and with dynamic word transition probabilities updated after each exchange.

At sentences level (table 2), the results show an improvement of 6%, with the use of probabilities. The results are 4% better when the probabilities are dynamically computed during dialogue: we can suspect that the richer the scripts are, the more important the increase should be.

The semantic recognition results shown in table 2 indicate that upper level knowledge is necessary to obtain satisfactory results. An increase of 16% is already observed with just the DTW algorithm.

The use of probabilities increases the message recognition accuracy to 96%. While dynamic probabilities give an additional improvement of 0.5%, but in regard to the

	DTW	with probabilities	with dynamic probabilities
Acoustic recognition	68.0%	74.0%	78.0%
Semantic recognition	84.0%	96.0%	96.5%

Table 2: Sentences recognition performance

	DTW	with probabilities	with dynamic probabilities
Acoustic recognition	96.7%	97.3%	97.7%
Semantic recognition	98.3%	99.5%	99.6%

Table 3: Words recognition performance

corpus size, these difference is not significant (we expect to find larger differences on more complicated tasks).

Timing results show that the recognition response time is reduced when dynamic transition probabilities are used, which allows the system to handle larger vocabulary and more complex tasks.

6 Conclusion

An oral dialogue system must be able to detect errors due to the speaker or to speech recognition, in order to correct them in an internal way and thus minimize the number of exchanges with the speaker. The system must also make all task-universe knowledge cooperate in the recognition process by providing predictions which dynamically restrict the recognition search space in order to increase the performance. Recognition experiments in a task-simulation environment indicate that the combined use of both dynamic probabilities and dialogue strategies allows the system to obtain performance equal 96.5% at the sentences level.

References

[1] K. Matrouf, F. Néel and J. Mariani "Système de dialogue orienté par la tâche: une application en aéronautique." *J.Acoustique 2 85-93* 1989.

[2] M. Minsky "A framework for representing knowledge," *in PH. Winston, the psychology of computer vision, Mc Graw Hill,* 1975.

[3] P. Falzon "Understanding a technical language, A schema-based approach." *Rapport technique NO 237 INRIA* 1983.

[4] JL. Gauvain "Reconnaissance de mot enchaînés et détection de mots dans la parole continue." *Thèse de troisième cycle,* Juin 1982.

[5] SR. Young, AG. Hauptmann, WH. Ward, ET. Smith, and P. Werner, "High level knowledge sources in usable speech recognition systems." *Communications of the ACM, Vol 32:183-194* 1989.

[6] S. Bornerand, F. Néel, G. Sabah, "Calcul dynamique de pondération sémantique dans un algorithme DTW," *submitted 18ème JEP* 1989.

[7] GM. Quénot, JL. Gauvain, JJ. Gangolf, and JJ. Mariani "A Dynamic Programming Processor for Speech Recognition", *IEEE journal of Solid-State Circuits Vol. 24, N. 2, April* 1989.

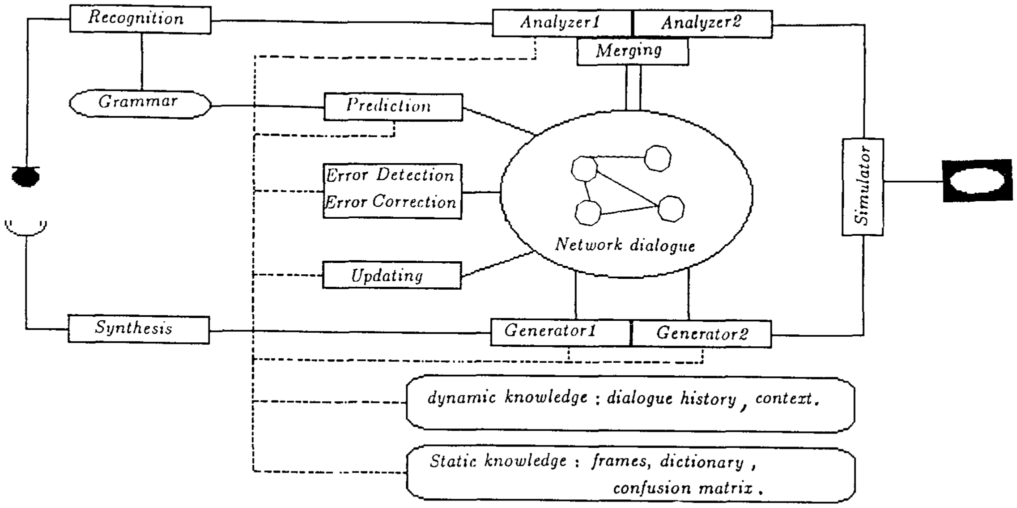

Fig 2 *Synoptic of the dialogue system*