LINGUISTIC PROCESSOR FOR A SPOKEN DIALOGUE SYSTEM BASED ON ISLAND PARSING TECHNIQUES

Aristomenis Thanopoulos, Nikos Fakotakis, George Kokkinakis
Wire Communications Laboratory, Electrical & Computer Engineering Dept.
University of Patras, 261 10 Patras, Greece.
Tel. +30 61 991 722, FAX: +30 61 991 855, E-mail: fakotaki@wcl.ee.upatras.gr

ABSTRACT

In this paper we present the Linguistic Analysis Component of a Spoken Dialogue System designed for robustness and flexibility. The dialogue takes place in the Greek Language through the public telephone network and is performed in two different applications. The analysis is based on Island Parsing, Pattern Matching and Frame-based Representation techniques. The main knowledge sources are a Semantic Network and Frame-Slot structures thoroughly connected with each other. Simple bigram grammar rules have been also used to assist the parsing process as well as to evaluate the recognition output.

10.21437/Eurospeech.1997-595

1. INTRODUCTION

Currently, Spoken Dialogue Systems development is not only a technological challenge but also a commercial demand. It requires the best design and cooperation between DSP and NLP technologies. In particular, the input analysis, that traditionally is obtained by the coupling of two corresponding modules, the Continuous Speech Recognizer (CSR) and the Linguistic Processor (LP), is the most crucial part of the whole process.

This paper presents the LP of a system with front end a CSR producing an output of a specific form (N-Best scored sentence hypotheses) with limited accuracy. The CSR input is obtained through the public telephone network. This degrades its output and, with spontaneous speech phenomena such as verbal edits, mid-utterance corrections, restarts, stutters, coughs, laughs, hesitations, ill-formed sentences, partial utterances, and out-of-vocabulary words, high flexibility and robustness is required from the LP part.

Some researchers argue that LP and CSR should be tightly coupled [1], so language restrictions could be applied at the recognition stage in order to increase recognition accuracy and speed. On the contrary, we chose a tandem topology in order to reduce computational complexity and increase the flexibility of the system (since we intend to use it, with domain modifications in several different applications). Moreover spontaneous speech usually doesn't follow the grammatical rules strictly. In our LP design we use expectations that emerge from the dialogue model and the current dialogue state. These Dialogue Expectations are applied very early in the linguistic analysis so that pragmatic-irrelevant solutions are not considered in the first place (cf.[2]).

The system is currently implemented in two different applications: an insurance company information task and a bank information retrieval task. The dialogue takes place in the Greek Language and is mainly machine-driven, i.e. the system addresses questions to the user, but without imposing him/her a strict way of answer. The task of the system is to extract the useful information and to continue the dialogue.

2. THE LANGUAGE FACTOR

One of the main facts that has guided the LP design is the special and crucial - for the recognition and understanding task - characteristics of the Greek Language. The latter include phenomena such as:

(a) Sentences constructed from phrases in a rather free order while phrases themselves are constructed from words in a strict order. This indicates that a traditional LR syntactic parser wouldn't be efficient enough since it would require either a complicated set of transformational rules or a lot of alternative syntactic rules to cover all potential phrase combinations within a sentence [3].

(b) A lot of short functional words exist without semantic content, e.g. articles, indefinite, personal and possessive pronouns (at least, the pronouns may be considered semantically empty at a limited domain). Each category has different forms in 3 cases, 2 genders and 2 numbers. Most of their forms are grammatically ambiguous, i.e. the word "του" (/tu/) may be an article (genitive case) or a possessive pronoun or a personal pronoun . The effect is twofold: i) In the (continuous) recognition stage, these words may easily be mistaken with other word syllables, replacing or replaced by them. ii) A full syntactic analysis would confront multiple solutions because of the inherent grammatical ambiguity and would probably be lacking of the desirable real-time speed.

EUROSPEECH '97
5th European Conference on Speech Communication and Technology
Rhodes, Greece, September 22-25, 1997

3. THE PARSING STRATEGY

For the robust linguistic analysis of misrecognized input several variations of LR parsing have been implemented, e.g. with a capability to skip words of the input in cases where the complete input sentence is not grammatical [4]. Another approach is to process parts of the input in order to trace word patterns constituting semantic fragments [5].

We have developed a similar but more general process with the latter, which assists the recognition evaluation of each sentence or phrase while keeping the run-time extremely short. The key for this is a novel ***Island Parsing*** algorithm that rapidly reduces the length (in number of tokens) of the sentence, traces the phrases and patterns within it, and produces few alternative solutions. That is, each sentence is considered as a list of tokens and the analyzer attempts to incorporate each token with its adjacent ones, in order to finally extract the sentence meaning from few comprehensive tokens.
Another strong reason for the island parsing-phrase spotting strategy is the nature of the Greek Language that allows a rather free phrase order within the sentence. Moreover it has been observed [4] that in spontaneous speech a phrase is usually uttered without breaks, hesitations or verbal edits internally and its syntactic structure is rarely violated, while the opposite happens to the boundaries of phrases. Also, experience in other systems [5,6] shows that island parsing is a more robust parsing process than LR parsing, since it can provide partial results and ignore meaningless chunks of the input. Finally the island parsing strategy can easily cooperate with a Word Graph output from SR, which is the most efficient interface between SR and LP for Speech Understanding, and will probably be used soon in our system.

4. THE PARSING ALGORITHM

The most fundamental part of our parsing algorithm is the pattern matching - phrase spotting procedure since the main task of the system is information extraction. This process is supported by a preprocessor that eliminates the words without semantic content and a post-processor that performs the island parsing algorithm.

At any stage of the whole parsing algorithm a scoring mechanism modifies the current "Understanding Score" of the sentence under process, according to its linguistic validity, semantic coherence and pragmatic consistency. Every process is performed in beam-search mode, i.e. every solution is to be considered further only if its understanding score reaches a certain percentage of the best so far.

The domain knowledge is represented in frame structure. Finally a frame-based representation is given and the most suitable candidate is selected, taking into account, in case of multiple solutions, the understanding and recognition score.

In the following we present a diagram of the LP (Figure 1.) and we describe the purpose and function of each module.

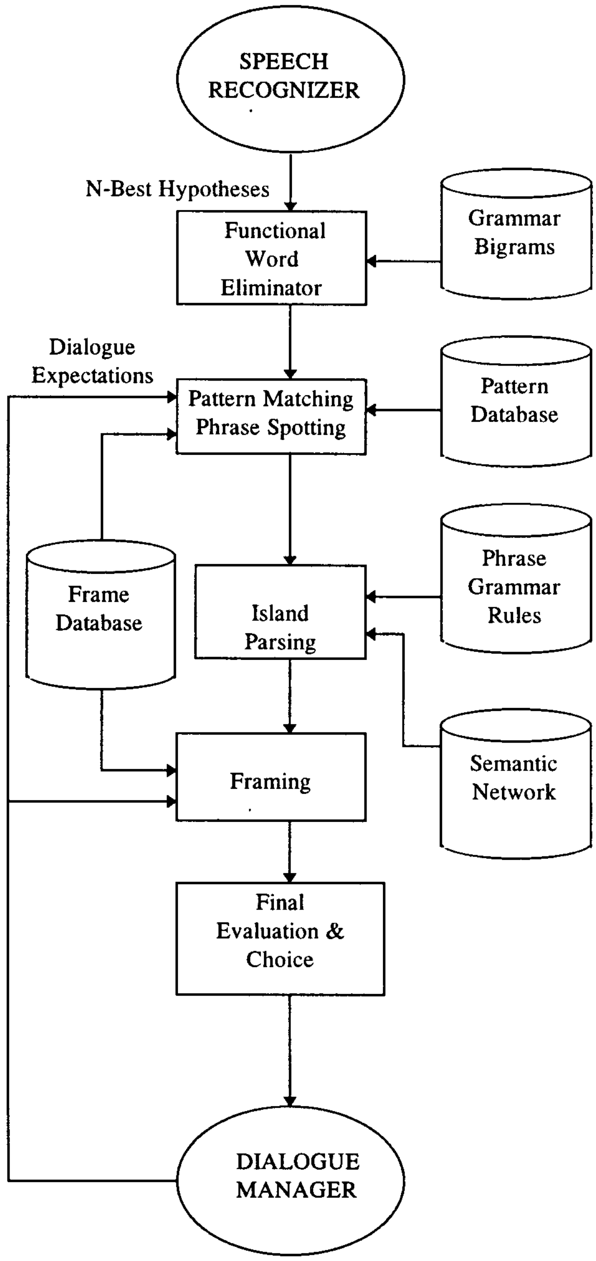

Figure 1: The LP modular diagramm.

4.1. Use of the Dialogue Expectations

During the whole process we make use, whenever it is possible, of expectations provided by the Dialogue Manager (DM), in order to save computational time.

The Dialogue Manager based on a dialogue model and on the dialogue history or simply on the last query of the system predicts what items of information or dialogue topics may be mentioned next. The current expectation determines the probable frame that must be activated. Here a frame-based reasoning mechanism [10] is operating in order to infer the possible subordinate (if any) frames that may be also of interest according to a frame hierarchy. The task of the following parsing process is to find fillers for these frames. For instance if the system asks:
- *"What service do you need?"*
the user may answer:
- *"I want to be informed about exchange rates"*, or, more focused,
- *"How much is the dollar value?"*
The first answer will be represented by the high-level frame {services} and its slot {exchange_information}, while for the second one the frame will be the immediately lower frame {exchange} and its corresponding filler the {US_dollar}. This frame inheritance is not globally allowed, but only in cases where the recognition and understanding will not be downgraded because of the multiple admissible solutions.

Since all possible frames have been determined in advance the parsing process aims to find fillers for the frame-slots of the selected frame(s), and finally the most plausible frame (if many) is selected. Therefore, the parsing process is Top-Down oriented and at every parsing step the dialogue expectations are applied as early as possible to limit the search space and save computational time.

The frame-slot representation has the advantage to allow partial results when trying to extract complex information. The rest can be acquired with the next question from the DM, which is focused on the missing information. For instance, for the system query:
- *"What is your birthday date?"*,
the expected frame {birthdate} has 3 slots: {month}, {monthday} and {year}. If, after the user's answer analysis, the two first slots have been identified with sufficient certainty but the third is ambiguous, the next system's question will focus on the year.

4.2. The Functional Word Eliminator

Every one of the N-Best sentences is transformed to a sequence of only semantically full words. All the Semantically Empty - Functional Words (SEFWs) are discarded in order to avoid the linguistic analysis of long sentences and to save computational time. Therefore we don't use the SEFWs for the understanding part but only for the recognition evaluation.

For this process we use bigram grammar rules to check the syntactic consistency of each of the SEFWs with its adjacent words. (For instance a personal pronoun is allowed to be followed or preceded by a verb). If such a consistency is found a "*syntactic validity score*" is attached to the corresponding (semantically full) neighbor word (the verb in our example), since this syntactic relation indicates a probable correct recognition of the word sequence. In either case the functional word is discarded from the sentence.

We must note here that we are not interested in number, case or person agreement between the constituents of the word-pair, since we can't count on accurate word-endings recognition, that carry this grammatical information.

4.3. The Pattern Matching process

During the pattern-matching procedure we are trying to trace the following possible types of patterns:

i) Patterns that carry highly concentrated information into specific string structures, such as dates, amounts, code numbers, etc.. The words that they consist of have a very focused and usually numerable contain. That is, they may be numbers, spelling letters, months of the year, days of the week, measurement units or special conjunctions (e.g. "slash", "dash", "of", etc.). These patterns are represented in the form of Recursive Transition Networks. The search is tightly directed by the dialogue expectations. This is performed on the primal level of the search using as heuristic criterion an extended syntactic feature that is attached to every word (e.g. number, weekday, etc.) to exclude irrelevant words.
ii) Characteristic utterances or phrases that are application dependent and we expect, on the basis of information extracted from Wizard-of-Oz experiments, that may be uttered at a certain dialogue point, after a certain system's query. For instance on the dialogue node that the system asks: *"Do you need another service ?"*, one possible form of a negative user answer will contain the utterance (or a fragment of it) : -*"I don't want (need) anything else."* Such utterances have been stored in a fixed form. Of course these sentences or phrases may not be uttered in this exact form, so a search is performed to find fragments of them in the input. In proportion to the completeness of the expected utterance a recognition evaluation score is given to the token into which the whole (partial or complete) utterance is incorporated.

4.4. The Island Parsing process

The objectives of the island parsing process are:

a) To spot the grammatical phrases (NPs, VPs, PPs) within the sentence. This is not only a necessary step towards understanding but also an efficient means for the evaluation of the recognition accuracy.

b) To reduce the length of the sentence in order to simplify it. Consequently the following meaning representation process becomes less complex and time-consuming.

c) To isolate meaningful chunks of the input from misrecognized and/or incomprehensible parts in order to finally obtain partial results from ill-formed input.

The island parsing process is performed step-by-step for the sake of robustness, i.e., in a first search along each sentence, word pairs that construct a phrase are found (e.g. {modifier+noun-->np}, etc.). Using a ***Semantic Network***, the semantic coherence of the phrase constituents is checked. If such a coherence will be found, the two tokens are incorporated into one with an increased understanding score and semantic content that of the most focused (of the two original tokens) or a compound one. A second search traces larger phrases (rules such as {noun+np-->np} are applied) using the same scoring mechanism and semantic checks, and finally a third search traces prepositional phrases ({preposition+np-->pp}, {prep+noun-->pp}).

Thus, robustness is preserved, since any failing matches leave the sentence untouched. Ungrammatical or misrecognized parts of the input remain as they are without a score improvement. In case that a higher-scored solution has previously appeared the current low-scored hypothesis is rejected. On the other hand, because of the absence of grammatical ambiguity among semantically full words in Greek, (common in English for example, as a word form may be verb or noun), this process doesn't generate many multiple solutions.

After this stage, the result is a list of sentence-hypotheses, each one consisting of few semantically full tokens and labeled with its initial recognition score. Each token is carrying a linguistic score and a structured (possibly complex) information content.

4.5. Frame Representation

For the meaning interpretation, the application-dependent knowledge is represented in a set of frames, each one containing a set of slots - labels for information items (cf.[4]). This frame-slot structure is directly connected to the semantic network.

Every surviving candidate solution is given a frame representation according to the expected frame(s). Then every semantic token of the sentence is checked if it is a slot filler or, using the semantic network, if it has a semantic connection with a frame or a slot entity. In any case a corresponding score is given. This score is negative in case that no semantic-pragmatic relation is found, indicating a probable misrecognition.

The final evaluation and the choice of the best and the most probable candidate frame representation is a process that takes into consideration the following criteria:

a) The degree of fullness and the quality of the fillers of the frame slots.

b) The number and importance of the remaining words that do not match the frame.

c) The current frame determined by the Dialogue Expectations.

d) The Linguistic Score of each phrase-token.

e) The Recognition Score of the sentence.

f) The number of each filler occurrence, in case that the hypothesis list includes several fillers.

5. CONCLUSION

We presented a Frame-based Island Parsing algorithm as the core of a Linguistic Processor that has been implemented in two Spoken Dialogue Systems operating in specific applications. We argued that this approach has several advantages concerning application flexibility, robustness against ill-formed or misrecognized speech input, real-time response and ease of implementation, and that it is mostly suitable for Syntactically Free Languages.

This linguistic processor is currently gradually tested in both applications, using a limited part of the dialogue which is steadily extended. The results confirm the above claims. Full implementation results will be available, as soon as the dialogue design is completed.

REFERENCES

[1] T.Kawahara et al: "Heuristic Search Integrating Syntactic, Semantic and Dialog-level constraints". In Proc. of IEEE-ICASSP, vol.2, pp.25-28, 1994.

[2] S. Young: "Use of Dialogue, Semantics and Pragmatics to enhance Speech Recognition". In Speech Communication vol.9, n.5/6, p.551-563, 1990.

[3] S.E.Michos, N.Fakotakis, G.Kokkinakis: "A Novel Method for Parsing Complex Sentences in Syntactically Free Languages", In Proceedings of the 6th International Conference on Tools with Artificial Intelligence, pp.253-259, 1994.

[4] M.Woszczyna et al: JANUS 93: Towards Spontaneous Speech Translation", In Proc. of IEEE-ICASSP, vol.1, pp.345-348, 1994.

[5] S. Issar, W. Ward: "CMU's Robust Spoken Language Understanding System". In EUROSPEECH 93, vol.3, pp.2147-2150, 1993.

[6] T.Kawahara et al: "Comparison of Parsing and Spotting Approaches for Spoken Dialogue Understanding". In Proc. of ESCA Workshop on Spoken Dialogue Systems, pp.21-24, 1995.