EFFICIENT AND ROBUST NATURAL LANGUAGE PROCESSING IN SIMPLE SPEECH APPLICATION DOMAINS

Manuela Boros

Bavarian Research Center for
Knowledge Based Systems (FORWISS)
Am Weichselgarten 7
D-91058 Erlangen, Germany
e-mail: boros@forwiss.uni-erlangen.de

Paul Heisterkamp

Daimler-Chrysler AG
Speech Understanding Systems
Wilhelm-Runge-Str. 11
D-89081 Ulm, Germany
e-mail: paul.heisterkamp@daimlerchrysler.com

ABSTRACT

Comparing spoken dialogue systems in commercial and research background reveals a great discrepancy in state-of-the-art systems. Whereas research systems tend to be very complex, allowing free and flexible dialogues using unrestricted speech, commercial systems are restricted to far less complex solutions such as menu-driven systems based on simple keyword-spotting, mostly doing completely without natural language processing. In our paper we present an approach to speech processing in dialogue systems based on linguistic phrase spotting, that allows for efficient and robust speech understanding in simple application domains. Misunderstandings or under-specification of user requests will thus be detected and complex dialogue strategies for clarification may be activated. In all other cases, the system may operate like a keyword-spotter. The proposed approach therefore is well suited for simple application domains such as information retrieval in commercial dialogue systems.

1. INTRODUCTION

Research in the area of interactive spoken dialogue systems has gained a lot of attention and effort in the past 10–15 years. For example, due to national and international funding (e.g. the DARPA ATIS-program in the US and SUNDIAL and its ESPRIT-successors in Europe) great advances in the field have been made. Current state-of-the-art research prototypes, even though restricted to limited domains, allow for flexible dialogues using spontaneous, rather unrestricted speech. Currently, the main focus of research lies on raising functionality and modality.

Some years ago, the first commercial dialogue systems emerged on the market. In contrast to research systems, commercial systems must absolutely satisfy a variety of demands, such as real-time processing, efficiency, easy maintenance, low development costs and re-usability. For this reason, commercial systems often are rather restricted in their functionality and flexibility, e.g. strictly menu-driven systems, where the user has to follow carefully the system's demands because it does not accept any input differing from its hard coded next processing step. In many cases, commercial systems completely do without natural language processing even where it would be beneficial. Rather, approaches such as keyword-spotting are preferred because of their apparent simplicity. However, developers of such systems recently become aware of the need of some kind of speech understanding capability, thus extending the simple keyword based approaches, e.g. the systems described in [3] and [4].

The phrase spotting approach, we propose in this paper, helps to bridge the gap between complex but often complicated and error-prone research systems and efficient and reliable but too inflexible and simple commercial systems. Based on the idea of partial parsing, the phrase spotter searches for the meaningful parts of an utterance, analyzes them and generates a unique semantic representation for the whole utterance. This semantic representation may be directly mapped onto the indices used to retrieve the requested data from the data base. In case no direct mapping can take place, either because of a misunderstanding or an under-specified user request, the dialogue module activates special dialogue strategies for recovery. Thus, our approach allows for the realization of simple but powerful systems, accepting unrestricted continuous speech.

For our first prototype we chose the domain of electronic operating instructions for cars: the instructions are accessible in the car using unrestricted spontaneous speech. This application domain is considered simple for the following reasons: Wizard of Oz experiments have shown, that despite the usage of unrestricted speech, only few different syntactic patterns are used. Also, typical dialogues are are rather short, and mostly the initial user utterance contains the complete query. The information to be requested may easily be stored and retrieved from a data base without involving complex inference mechanisms.

2. SYSTEM OVERVIEW

We implemented a state-of-the-art spoken dialogue system comprising the modules word recognizer, linguistic processor, dialogue manager and speech synthesizer. The overall architecture of the system is depicted in figure 1.

The word recognizer delivers a word graph, that is analyzed and assigned a semantic representation by the linguistic processor. The analysis is done with the aid of the linguistic knowledge stored in the do-

ESCA Workshop on Interactive
Dialogue in Multi-Modal Systems (IDS-99)
Kloster Irsee, Germany
June 22-25, 1999

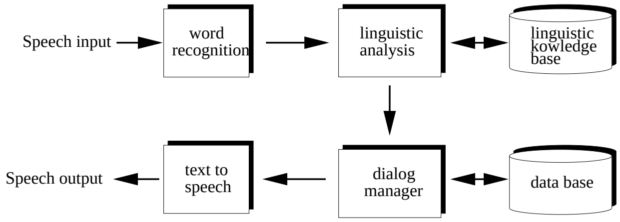

Figure 1. Architecture of the system.

main dependent lexicon and grammar. The semantic representations are passed on for interpretation in the actual dialogue context to the dialogue manager which also guides the dialogue flow, the data base queries and system answers. We made some important changes to the control flow by reducing the dialogue function to a minimum. The semantic representations that specify a data base entry are directly used for indexing the data base. Therefore, the semantic representations delivered by the parser can be directly matched against the data base indices and do not need to be interpreted and transformed into a data base query by the dialogue manager. The default dialogue strategy can therefore be kept quite simple: if the user provides a complete information request (complete meaning, that the semantic representation of the utterance matches an index in the data base) the corresponding information is directly output to the user, which in the addressed domain means, that an audio file is played. In case of misunderstandings or under-specifications, i.e the semantic representation, generated by the phrase spotter, does not match any data base index, interpretation in the dialogue manager takes place and dialogue strategies such as asking for specification, clarification or confirmation are activated.

There are two major advantages of this strategy: First, due to the mapping of semantic representations on data base indices interpretation is only needed in case of a mismatch. Second, dialogue strategies restricting the user will only be activated when necessary, thus allowing for unrestricted speech most of the time.

2.1. The Linguistic Processor

The linguistic phrase spotter, representing the core of the linguistic processor, is based on the idea of partial parsing. The partial parsing technique is a powerful means to cope with ill-formed input, which is an essential demand in speech processing. Several different approaches to partial speech parsing have been successfully employed in spoken dialogue systems, such as the ones described in [1] and [2]. Our approach goes beyond these 'traditional' types of partial parsers: it provides partial analyses for sentences, restricted to their semantically relevant parts, **and** combines these partial analyses to one single analysis, according to pre-defined patterns of valid sequences of partial results. These valid sequences are defined in terms of *meta-rules* in the grammar and are based on typical patterns found in the domain's user utterances (cf. section 4.1). Phrase spotting is performed in two steps. Step one comprises the analysis of relevant parts of the utterance, step two comprises the search for *key-phrase patterns*, in order to achieve a complete analysis.

Lexicon and grammar constitute the system's linguistic knowledge base. The lexicon contains every word, the system has to know in order to understand an utterance correctly. Each lexical entry is assigned a complex feature structure containing morphologic, syntactic and semantic information on the respective word form. The grammar contains rules for the combination of words or phrases to larger expressions. As only partial analysis is performed, no full grammar has to be coded, but a collection of grammar fragments. These grammar fragments cover all the relevant parts of typical user utterances. Relevant meaning here, that the respective parts of an utterance are crucial for its correct interpretation by the system. For example, in the German sentence *Wo ist hier der Lichtschalter?* (Where is here the light switch?), only the interrogative *wo* (where) and the noun phrase *der Lichtschalter* (the light switch) are relevant. The verb may be neglected, because it is implicitly given by the interrogative (cf. section 4.2). The relevant parts can be identified from sample data e.g. collected by Wizard of Oz experiments. Additional, for each utterance the sequence of these relevant parts is determined in order to set up appropriate grammar rules. These rules are called meta-rules because they do not necessarily conform to standard grammar. The grammar is defined in terms of a context-free phrase-structure grammar.

3. THE PHRASE-SPOTTING ALGORITHM

Based on the island parsing technique, we implemented the phrase spotter in terms of an agenda-driven chart parser ([5]). Island parsing is a successful approach to achieve robustness in speech processing: Starting at arbitrary edges in the chart, the parser searches to the left and right until no further expansion of chart edges can take place. If the initial islands are chosen carefully, this approach makes sure, that the most important parts in an utterance are analyzed at the very first beginning. The parsing process is guided by the agenda: its entries are the initial chart edges (*seed entries*), as well as pairs of edges to be combined, inserted during parsing (*non-seed entries*). Agenda entries are scored and sorted accordingly, thus in each parsing step, the most promising entry (the one with the best score) will be chosen for further processing.

Phrase spotting takes place in two different processing levels: during level 1, the relevant parts of an utterance are determined and analyzed. This may result in a complete parse, in case of utterances containing just one single word or, more general, one single relevant part. In case, that the parsing level 1 does result in several partial analyses due to several relevant parts in the utterance, a second level of processing takes place. In level 2, the partial analyses are combined to one single chart edge by using the meta-rules in the grammar. Thus, after parsing of level 2, one unique semantic representation of the utterance has been generated.

3.1. Parsing of level 1

Before parsing, the chart and agenda are initialized with the hypotheses in the word graph. As only parts of the utterances are covered by the lexicon, many hypotheses will have no corresponding lexical entry. In order to avoid gaps in the chart due to unknown words, for each unknown word a *dummy edge* is inserted into the chart. Dummy edges keep the information on the acoustic score as well as start and end node of the original hypothesis, but do not include any linguistic information. Thus, the original structure of the word graph is copied to the chart. As example consider the German sentence *wo ist der Lichtschalter* (where is the light switch). For its correct interpretation only the interrogative *wo* (where) and the noun phrase *der Lichtschalter* (the light switch) are important. Therefore, the verb is not covered by the lexicon and is simulated by a dummy edge in the chart, as shown in figure 2.

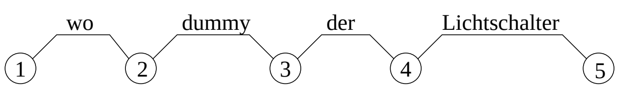

Figure 2. Initial chart containing a dummy edge.

Starting with the best scored agenda entry, the parser will follow the following steps until the agenda contains no more entries:

1. Take the best scored agenda entry E from the seed agenda.
2. If E is a seed entry proceed to 3, if E is a non-seed entry proceed to 4.
3. For each chart edge adjacent to E: look for rules, that can be applied to both, generate an agenda pair for both and sort it into seed agenda; proceed to 1.
4. For each grammar rule given in E: apply this rule to both edges, insert the new edge (if the rule can be applied) into the chart, generate new agenda pairs for this new edge and insert them into seed agenda; proceed to 1.

For each new chart edge, a new score is computed based on the acoustic score of the original edges. This algorithm makes sure, that new chart edges get a better score than the original ones.

During level 1, dummy edges are not taken into account. Thus, if the chart contains dummy edges, only partial analyses can be found. In the current example, the parser will generate one new chart edge for the noun phrase *der Lichtschalter*, whereas the initial edge *wo* won't be expanded due to the adjacent dummy edge. Figure 3 shows the resulting chart.

In case no full spanning edge has been generated during parsing level 1, parsing level 2 is started.

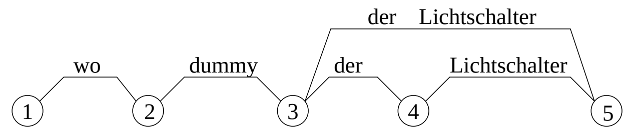

Figure 3. Chart after parsing of level 1.

3.2. Parsing of level 2

Parsing level 2 aims at combining the partial analyses found in level 1, to one full-spanning edge representing the analysis of the complete graph. At this stage, dummy edges have to be considered, because they provide information on which edges are part of the same path through the graph. Only edges that are part of the same path may be combined with each other.

The parsing algorithm is the same as for level 1. The chart from level 1 is kept, whereas the agenda has to be re-initialized. Here, not all chart edges are asserted into the agenda, but only the n best scored. The reason is, that initialization with all edges would result in the same agenda as for level 1 **plus** the there generated new edges. Consequently, every parsing step from level 1 would be repeated. Instead, we only want to use the maximal edges, i.e. the partial results of level 1. As larger chart edges achieve a better score than the edges they span, the agenda entries are chosen according to their score. The number of 5 proved to be sufficient for n.

During parsing, edges separated by dummy edges may be combined with each other if the grammar holds an appropriate rule. Here, the meta-rules are used, that define the valid patterns of relevant parts in utterances. Figure 4 shows the resulting chart for the example sentence after parsing of level 2.

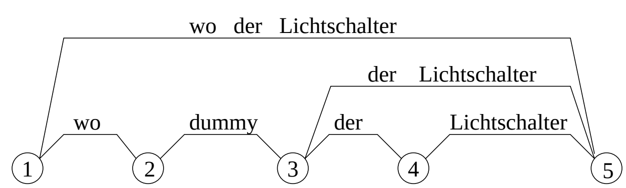

Figure 4. Chart after parsing of level 2.

As the grammar contains a rule to combine an interrogative with a noun phrase, the dummy edge has been crossed and the full-spanning edge *wo der Lichtschalter* has been generated. In parallel, the respective semantic representation has been generated.

4. PRELIMINARY RESULTS

With our first prototype, we addressed the domain of electronically available operating instructions for cars. Located in the car directly, the operating instructions are accessible via unrestricted spontaneous spoken queries. The requested pieces of information are extracted from the data base, and displayed to the user. In total, the data base contains 289 different topics, each of which comprises at most three different types of information: warnings, definitions/indication

of place and general operating instructions, leading to a total of almost 800 data base entries.
Each entry in the data base is assigned a unique index. The indices are identical with the semantic representations. This means, that for each type of information, a unique semantic representation has to be defined, as well as for each topic. The following example shows the semantics for the indication of place of the hand break in the frame-like formalism we use for semantic representation. A corresponding user utterance may be *Wo ist die Handbremse?* (Where is the hand-break?).

```
type: request,
themode: where,
thetheme: type: hand_break.
```

The requested topic is defined using the keyword `thetheme`, the type of information for that theme is defined using the keyword `themode`. The type of the whole index being `request` makes clear that this is a data base index, not to be confused with other kinds of user utterances such as greetings or commands.

4.1. Wizard of Oz Experiments

By Wizard of Oz experiments, typical user utterances have been collected. This data revealed three interesting observations: First, a rather small number of different syntactic patterns can be identified, that are used by almost every user to formulate his requests. Second, once a pattern lead to success, i.e. the requested information was provided to the user, he would never change his strategy again until a misunderstanding occurs. Third, the user utterances are quite short, in many cases, the user will just say the name of the topic he wants to know something about. Based on these observations, the syntactic patterns, the system must cover in order to understand the user's queries, could be defined.

4.2. Linguistic Coverage

For each of the above described types of topics, stereotypical user queries could be identified. Table 1 shows these syntactic patterns.

Type	Pattern
Definition	Was ist $THEME$ (What is $THEME$)
Definition	$THEME$
Operating Instructions	Wie funktioniert $THEME$ (How works $THEME$
Warnings	Was muß ich bei $THEME$ beachten (What do I have to mind with $THEME$)

Table 1. Patterns for information request.

$THEME$ stands for any expression that is associated with one of the topics in the instructions.This may be a single noun, as well as complex phrases (e.g. paraphrasing expressions for nouns). Often, in order to ask for definitions, only the topic was uttered. Questions concerning the operating instructions may as well contain another verb than *funktioniert* (works), in any case, the interrogative *wie* (how) and the theme has to be included. Table 2 shows some of the possible realizations for $THEME$:

Description	Example
Simple Nouns	Licht, Gang, Tank *light, gear, tank*
Compound Nouns	Lichtschalter, Tankanzeige *light switch, fuel display*
Noun Phrases	Lösen der Handbremse *loosening of the hand-break* Position des Lenkrads *position of the steering wheel*

Table 2. Examples for complex noun phrases.

These nouns and noun phrases make up most of the lexical entries and grammar rules. The current grammar, covering most of the themes and their paraphrases, comprises 12 grammar rules, only 4 of which are meta-rules. The lexicon is quite large with 584 entries, most of them nouns. First experiments run with real word graphs proved the phrase spotter to be efficient and robust even for huge graphs.

5. CONCLUSION AND FURTHER WORK

In our paper we presented a robust, efficient and powerful approach to spoken dialogue systems. The proposed phrase spotter allows for the partial analysis but complete semantic representation of utterances, thus making a bypass of complex dialogue functions possible. Just in case of misunderstandings or underspecification the complex dialogue strategies are invoked. In the near future we plan to evaluate the system further with a greater variety of test users.

REFERENCES

[1] D. Albesano, P. Baggia, M. Danieli, R. Gemello, E. Gerbino, C. Rullent. A Robust System for Human-Machine Dialogue in Telephony-Based Applications. In *International Journal of Speech Technology*, Vol.2, pp. 101-111, 1997.

[2] H. Aust, M. Oerder, F. Seide, V. Steinbiss. The Philips automatic train timetable information system. In *Speech Communication*, Vol.17, pp. 249-262, 1995.

[3] Kono Y., Yano T., Sasajima M. BTH: An Efficient Parsing Algorithm for Word-Spotting. In *Proceedings of ICSLP'98*, Sidney, 1998.

[4] Lin B., Chen B., Wang H., Lee L. Hierarchical Tag-Graph Search for Spontaneous Speech Understanding in Spoken Dialog Systems. In *Proceedings of ICSLP'98*, Sidney, 1998.

[5] K. Mecklenburg, P. Heisterkamp, and G. Hanrieder. A robust parser for continuous spoken language using Prolog. In *Proceedings of the Fifth International Workshop on Natural Language Understanding and Logic Programming (NLULP 95)*, pp. 127–141, Lisbon, 1995.