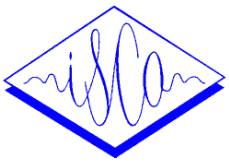

2nd International Conference on
Spoken Language Processing (ICSLP 92)
Banff, Alberta, Canada
October 12-16, 1992

Linguistic Modelling in the Context of Oral Dialogue * †

Gerhard Th. Niedermair

SIEMENS AG
Otto-Hahn-Ring 6
8000 MUNICH 83, GERMANY.
nie%zeisig@zfe.siemens.de

Abstract

This paper describes the integration of linguistic knowledge into the German prototype **SUNGerm** of the European SUNDIAL project. The goal of the project is to construct a telphone based real time system for oral inquiries into a database of intercity train schedules[1].
We will describe the system set up and the interaction among the speech recognition, linguistic analyser, and dialogue modules. We suggest how these modules can effectively interact to improve recognition and understanding through language models dependent on the context of the dialogue.
Additionally, we outline the linguistic analysis concerning phenomena frequently encountered in oral dialogue. We describe how these phenomena are captured in a semantic interface language (SIL) which is uniform for the different language prototypes within the SUNDIAL project.

10.21437/ICSLP.1992-211

System Set-up

Late in 1991 the German prototype of the SUNDIAL project had been fully implemented. It integrates modules that are the result of a common effort of different partners of the SUNDIAL project. The task domain of the system is inquiries on time schedules of intercity train connections. The application has a vocabulary of around 1100 single word forms including city names and the like. The prototype presented here uses an acoustic front end that has been provided by a German project partner[8][1]. At the time of demonstration the prototype's input characteristics were multi-speaker mode and high quality speech. Current plans for the next prototype include speaker independence and telephone speech quality. System output is fully synthesized speech. The dialogue is conducted in a speech-in speech-out mode with no terminal involved.

For management of the task oriented dialogue, the German prototype integrated an adaptation[2] of the generic dialogue module which had been implemented by a common effort among several partners in the project[2]. The dialogue has a strategy of mixed initiative where the user, after an introductory remark by the system, may formulate his information request. The system confirms the parameters it has found and which are of relevance for the completion of the task. It then continues to ask for additional parameters until it has sufficient information to start a database query. The result of this is presented and repeated orally to the user together with an offer for further or alternative connections. An example of such a dialogue[3] is given below:

S:	*here is the automatic intercity train service. which information do you request*
U:	*i would like to go from hamburg to munich*
S:	*you want to go from hamburg to munich*
U:	*yes exactly*
S:	*on which day would you like to travel*
U:	*the 25th of april please*
S:	*at what time would you like to travel*
U:	*... well sometime around eight ok*
S:	*you want to travel on the 25th of april between 7.45 and 8.15 in the morning*
U:	*that's it*
S:	*you depart from hamburg 8.01, hamburg 8.01, and arrive in munich at 15.10, munich 15.10. would you like further connections*
U:	*no thanks bye bye*

Example of train inquiry dialogue in SUNGerm

To improve the recognition quality and to increase, at least potentially, the understanding rate of the linguistic module, not only linguistic constraints should be applied for the recognition of word sequences but also constraints derived from dialogue situations.

Abandoning the often used sequential generate-and-test architecture, we employ bidirectional interfaces between three modules: the acoustic recognizer (AR), the linguistic module (LM) and the dialogue module. The LM is able to receive messages from the dialogue module (DM) that, given a dialogue situation tries to predict the next move on the part of the user[3]. If we are able to feed that knowledge down to the acoustic recognition module, then we should be able to improve recognition and understanding.

Dialogue Dependent Bigram Models

In our architecture we feed the knowledge on the state of the dialogue, which the LM receives from DM down to the AR by means of specialized language models.

The current AR uses a stochastic bigram model based on categorial bigrams[4] to support the recogniton. First, a general model was trained on examples for the whole task, either taken from simulations [5] or from example sentences from the staff. This general language model has a perplexity of 106. This was the lowest perplexity that we could achieve through appropriate intellectual classification of the words.

*The research reported here is partly funded by the CEC, DG XIII, in the ESPRIT programme as project 2218 SUNDIAL.The partners in this project are CAP GEMINI INNOVATION, CNET, CSELT, DAIMLER-BENZ, ERLANGEN University, INFOVOX, IRISA, LOGICA, POLITECHNICO DI TORINO, SARIN, SIEMENS, SURREY University.

†The author would like to thank his colleagues in the SUNDIAL project, who contributed to the work described here: G.Bakenecker,H.Tropf, P.Witschel, U.Ziegenhain, T.Kuhn,P.Heisterkamp

[1]The University of Erlangen

[2]this was done by Daimler Benz

[3]here the English translation of the German dialogue

However, depending on the preceeding question of the system, our oral dialogues exhibited a more or less predictable variation in complexity and content of the user's utterances. This fact should be exploited during recogniton.

We trained dialogue step dependent language models following the observation that the utterances between different users were rather similar within a certain dialog step but very different in vocabulary and structure between different steps. We grouped the corpora into 14 different classes depending on the semantic and situational context of the dialogue. The training material and test material were extended by spontaneous written utterances in particular dialogue situations. The corpora sizes are displayed in table 1. For these corpora, corresponding stochastic bigram language models using syntactic, morphologic and semantic categories were generated. For a more detailed description see [4].

Table 1 displays an overview of the different contexts in the train schedule enquiry domain for which the models were generated: b_0 stands for the general language model comprising all different dialogue situations , b_1 to b_14 are the dialogue con-

LM No	preceeding system question situational context	train. corp. utter.	test corp. utter.	test set perpl.
b_0	whole task domain	1947	458	106
b_1	hello, what info do you need	350	75	56
b_2	**at what time** do you want ...	144	22	49
b_3	**when** do you want to travel	157	22	48
b_4	do you want the **next train**	150	25	42
b_5	asking for **confirmation**	200	60	45
b_6	**where** do you want to go	100	35	69
b_7	which city do you want **to pass**	99	19	85
b_8	you want **earlier/later** connect.	160	40	64
b_10	**on which day** ... to **leave**	152	30	71
b_11	**at what time** ... to **arrive**	150	25	69
b_14	**thank you, good bye**	35	25	65

Table 1: *Dialogue context dependent language models*

text dependent language models. The context string indicates the preceeding system utterance which established the particular dialogue context. The 11 different models are trained on the complete lexicon. Table 1, most importantly, shows the different test set perplexities for each model. Each model still models the whole vocabulary, just with varying probabilities. Consequently the size of each context dependent model stays the same, only the transition probabilities between the category-classes vary from model to model. In this way the model does not a priori exclude any possible user utterance from being recognized[4] at any given time during the dialogue.

Other than word pair grammars or cascaded predictions [6] which gradually loosen their constraints when the utterance cannot be analysed, the stochastic dialogue dependent models just assign a higher priority to an expected utterance. An unexpected utterance can be analysed as well, assuming that its acoustic score can outweigh the rather low probability.

Table 1 shows that for the dialogue dependent models we get a perplexity improvement from 19% up to 60% in comparison to the perplexity of b_0. This decrease in perplexity should of course show up in the recognition figures of the AR. Recognition accuracy improved by 0%-16%. The AR was tested on the same sentences that were also taken for the calculation of the test set perplexity.

[4]as e.g. the sometimes used context dependent word pair grammars do, however,they require on average less computing power

In order to invoke the appropriate LM at the right time, the AR, when called, is provided by the LM with a parameter that tells it which language model to use for the next user utterance. The models themselves are static data structures inside the AR.

The LM derives this knowledge from the predictions which, for each utterance, are passed from DM to LM. A prediction is derived from the state of the DM and the type of utterances that the system last made. The prediction contains (besides system internals) the following information:

(1) The kind of dialogue acts that are expected for the next user input. Typically predicted acts are e.g.: open-request, correction, confirmation, wh-answer, etc..

(2) The dialogue history provides information on the topics that have been addressed last, either by the user or the system. They are passed down as semantic types like "time", "date", etc.

From this information we extract the relevant characteristics which allow us to associate a particular language model with the current dialogue situation, and pass an identifier for the model to the AR. If no context dependent class can be associated with the prediction, the general language model b_0 is used as a default.

Syntactic Analysis

At the interface between AR and LM the AR currently outputs the best scoring sequence of words or alternatively a graph of scored hypotheses. In the version described here we worked with the best scoring hypothesis. Preliminary experiments with a test corpus of 100 sentences from the SUNDIAL domain have shown that the correct utterance had the best score in about 70% of the cases. With our current parser we were able to gain additional 7% correctly analysed sentences when the full hypotheses graph was explored. This, however, required a large expense of computing time. In order to exploit this margin we are currently trying to employ more efficienct search strategies to achieve faster processing of hypotheses graphs.

For the demonstration system, the acoustic output is analysed with a modified Tomita parser which uses an augmented phrase structure grammar. The grammar is designed to cover typical oral dialogue phenomena such as partial sentences, one-word utterances and communicative phrases as well as dialogic particles, politeness phenomena, colloquial speech and combinations thereof.

Partial sentences: In the grammar, partial sentences are given the same status as sentences. As are isolated noun phrases, prepositional phrases and adverbial phrases. In this they share the same characteristics as full fledged sentences especially with respect to their ability of being surrounded by dialogic particles like *" yes"*, *"exactly"*, as well as politness particles like *"thanks"*, *"please"*,etc. Several features regulate the sequental order of the particles. One-word utterances, which frequently occur, can be either dialogic particles, like *"yes"*, or simple elliptical answers to wh-questions like: *"munich"*, after the user was asked for e.g. a destination. Particles have sentence status as well. Given the current complexity of the task, there has been no need to 'justify' such hypotheses by an 'ellipis resolution' component.

Multiple Utterances: Frequently in spoken language an utterance may contain more than one sentence. Instead of leaving it to the parser to restart after one sentence (but not yet the entire input string) has been processed or to postprocess the chart after the complete analysis of the input chain and collect all possible sentences or sentence-like phrases, we designed rules that grammatically describe the input string as a sequence of sentences in a recursive manner. The nonterminal element describing this is the 'Utterance'. Utterances may consist of one or more sentences, each of which can, in the above manner, be enclosed by different

particles. Together with the rules for partial sentences, this allows for a rich variety of constructions especially when particles and politeness phrases are involved and accounts for many phenomena of spontaneous speech, without overloading the grammar with a large number of different rules.

Dialogic particles have sentences status. Hence in utterances like:

"yes to munich please at five"

the nonterminal 'Utterance' would span the entire input which in this case is regarded as an utterance consisting of three seperate 'sentences', which are *"yes"*, as a dialogic particle, confirming the system question, *"to munich please"*, as an isolated PP with an attached politeness particle, and *"at five"* again as an isolated PP giving additional information. In the same way even more complex utterances like:

"oh yes exactly to munich yes please at five ok"

can be interpreted correctly. The syntactic analysis is backed up by a mechanism that can check semantic constraints between constituents like verb and noun phrases, prepositional phrases and adverbial phrases. Also inside constituents semantic agreement is checked like between nouns inside appositional constructions, or between the type of preposition and the semantic head of the PP, or between headnoun of NP and its attributive PP's. These semantic agreement tests are part of the grammar test and are carried out on the basis of a semantic network that models the concepts of the domain and their surface semantic dependencies. For a description of the mechanism see [7].

This syntactic semantic anaylsis was tested on a large corpus of transcribed utterances. This coorpus is a subset of the corpus for training the language models. Table 2 shows the results of different test corpora. An error in these cases means that the sentence was not parsed correctly because of an incorrect grammatical description or too severe semantic constraints. This also indicates that explicit and presriptive intellectual modelling of the semantic domain is generally too rigid and needs, in future, to be combined with probabilistic methods.

no of sent.	corpus	error rate
100	Sundial test set	2.0%
239	Sundial Germ	0.6%
458	Sundial Dialogues	0.8%
240	4x60 answers	12.9%

Table 2: *Parsing Results for Sundial corpora*

The 'Sundial Dialogues' is a collection of typical dialogues as they may be conducted in the current system. The errors in these dialogues are due to some rare syntactic semantic phenomena which have not been modeled yet. More telling is the figure for the '4x60 answers'. Several subjects, who were familiar with the vocabulary were asked to answer questions, as they are currently asked by the dialogue manager. This collection of sentences had not been seen before neither by the system nor the developers.

With these corpora we achieved average parsing times of:

Syntactic analysis alone:	1.8 sec	ranging from 0.3 sec to 5 sec
Including semantic constraints:	2.1 sec	

Table 3: *Parsing Speed for Implementation in Lisp on TI-Explorer II*

When the AR was tested in connection with the LM we achieved the following results on the 'Sundial test set'. (As we use the best hypothesis only, the parsing times are the same as above)

Status	no. of sent.	in % compared to 98
Grammatic.correctly treated:	98 sent.	
Correctly understood:	86	= 87%
At best scoring place:	68	= 70.4%
Including Graph search:	76	= 77.5%
No correct alternative:	12	
With substitutions:	10	

Table 4: *Sources of Errors in Parsing*

In table 4 the 'No correct alternative' figure shows the number of sentences where no correct sentence could be found, neither in the best scoring hypotheses nor in the graph. These cases could only be overcome by more robust parsing techniques which are able to analyse sentences that are outside of what is described grammatically. The 'With substitutions' figure shows the number of sentences where one or more words were substituted for others, but which still rendered a syntactically and semantically correct sentence. Since these are errors that cannot be attributed to nor remedied by the linguistic processor, the number of cases in which the linguistic processor worked correctly in isolation is the 'Correctly understood' figure.

Semantic Representation in SIL

The generation of the semantic representation in SUNGerm is based on SIL (Semantic Interface Language), which includes a semantic representation language and a hierarchy of semantic types with associated roles. It has been developed within the SUNDIAL project as a common effort between several partners.

The LM of the German demonstrator outputs a SIL representation of the best scoring sentence. The first available reading is chosen if the structure of the sentence is ambiguous.

SIL defines a hierarchy of types for the train and flight schedule domain, common to all partners. This guarantees that a uniform interface for all languages will be presented to the dialogue manager, which internally uses the same type hierachy. For each semantic type in this hierarchy the roles, the types can take as well as the possible values (at the leaves of the hierarchy) are specified. The roles are further restricted by the types they can be filled with. E.g. the type 'go' may have, among others, the roles 'thesource' and 'thegoal', which may both be filled by semantic objects of the type 'location'. Each semantic object receives its own id_no in order to allow cross-references among parts of semantic representations and to the syntactic description. E.g. a partial sentence like:

"...go to munich ..."

would be represented as:

```
[id:_,type:go,
  thegoal:[id:_,type:location,
           thecity:[id:_,type:city,
                    value:munich] ]]
```

Semantic representations in SUNGerm are built up on a rule by rule basis after the syntacic semantic parse has come up with a solution. It is more efficient to apply the rather 'expensive' SIL construction rules only for those edges in the syntactic tree that have survived during the analysis. For each rule in the syntactic semantic grammar there exists a 'SIL structure' rule, which is

corefferenced by number. All the syntactic and semantic features of the constituents, as they have been set and percolated in this rule during syntactic semantic analysis, are available to the corresponding 'SIL structure' rule. The 'SIL structure' rule consists of a construction part, a feature percolation part, and a part that can check the conditions on all features of the rule's constituents. In the construction part, the up-to-now SIL representation of the constituents, together with features like 'definiteness' or 'case' which may play a role for proper SIL represenations, are handed over to a construction function that generates the resulting SIL representation. A function like e.g.:

```
build_sil_vp_pp(V_rep+,PP_rep+,Case+,Prep+,Newrep-)
```

called by the rule that binds an PP to a VP would take the up-to-now represenation of the PP: `[id:_,type:city,value:munich]` and the VP: `[id:_,type:go]` , the case of the PP: (Dat), and its enclosed preposition: ('to'), which are feature values of these non-terminals, and construct with the help of 'SIL transformation files' the appropriate new path (see above), which becomes a feature of the left hand side of the rule. In such a way the represenation is built up through the entire tree in a bottom up fashion. When new verbal arguments are attached to the verb the new path tree may share parts with the current np-representation, like in:

"...go to munich to the main station"

The type 'go' is defined with a role 'thegoal',which is type-restricted by type=location. 'Location' again has as potential roles: 'thecity' and 'thestation'.The representation of 'go to munich'(see example above) is combined with the representation for 'to the main station', which is:
`[id:_,type:station,value:central,modus:[def:the]]`
This is done by first attaching the constituent to the verb 'go' and creating a full path for the resulting combination, which is structurally similar to the above one for 'to munich', and subsequently unifying the two SIL trees, resulting in:

```
[id:_,type:go,
   thegoal:[id:_,type:location,
            thecity:   [id:_,type:city,
                        value:munich]
            thestation:[id:_,type:station,
                        value:central,
                        modus:[def:the]] ]]
```

Represenations of utterances that consist of more than one sentence can also be built up on a rule by rule basis. The rules for combining the sentences are paralleled by the 'SIL structure rules' for the generation of the respective SIL represenations for multiple utterances. Also in this respect, this approach is very convenient in comparison to 'postprocessing' approaches.

Conclusion

We have described the prototype architecture and implementation of SUNGerm, in particular the linguistic processing and the application of linguistic knowledge, at the level of AR. We have presented the results of experiments that show that the perlexity for the recognition task can be reduced considerably through the use of context dependant stochstic bigram models . Although the size of the training material for these dialogue dependent models may be considered too small to deliver reliable results, there are two factors that can counteract this. First, the variability in this kind of task is small; therefore a smaller amount of training material may be sufficient to start with; second, the training material (which is hard to come by anyway for any kind of dialogue application) may in future versions be used as a starting point to initialize the system models. If thereby a workable system can be achieved, which can be put to the test with more users, then the newly received input data can be used to gradually improve the probabilities of the model and therefore make it more reliable.

We have outlined some of the phenomena that we have to handle in our task oriented spoken dialogue, how they are dealt with in an augmented phrase structure grammar, and how they are represented in a common representation language SIL. Finally we have shown that with a 'best hypothesis' strategy, satisfactory results can be achieved. However, there is still a wide margin of cases that could be covered by linguistic analysis if graph searches could be speeded up considerably and if robust techniques could be employed to restore sentences in which only one or a few words are incorrect. The same techniques could take care of ungrammaticalities that often occur in spoken dialogue. Both lines will be investigated for the futher demonstration prototypes.

References

[1] Peckham J., "Speech Understanding and Dialogue over the telephone: an overview of the ESPRIT SUNDIAL project." *Acoustic Bulletin*, 1990.

[2] McGlashan, S., N. Gilbert, N. Fraser, E. Bilange, P. Heisterkamp, and N. Youd, "Dialogue management for telephone information systems." in: *3rd Conference on Applied Natural Language Processing. ACL.*, Trento, Italy, April 1992.

[3] Andry F., "Static and dynamic predictions: A method to improve speech understanding in cooperative dialogues." in: *Proceedings of the International Conference on Spoken Language Processing*, Banff, Canada, Oct. 1992.

[4] Witschel P., Niedermair G.T., "Experiments in Dialogue Context Dependent Language Modelling." to appear in: *Proceedings of the Konvens Conference*, Erlangen, Oct.1992

[5] Kritzenberger H., "Zustandsbeschr. der Benutzereingaben zum DICOS-Bahnauskunftssystem." Universitaet Regensburg, FG Linguistische Informationswissenschaft, DICOS-Arbeitspapier No.11, 1990

[6] Young S., "The MINDS System : using context and dialogue to enhance speech recognition." in: *Proceedings of the DARPA Conference*,1989, pp.131-136

[7] Niedermair G.T., "The Use of a Semantic Network in Speech Dialogue." in: *European Conference on Speech Communication and Technology.* 1989, pp. 026-029

[8] Kuhn T.,Niemann H., Schukat-Talamazzini E.G., Eckert W., Rieck S., "Context-dependent modeling in a two-stage HMM word recognizer for continuous speech." in: *EUSIPCO'92-Proceedings of the EUSIPCO 92*,1992