A Speech Understanding and Dialog System with a Homogeneous Linguistic Knowledge Base

Marion Mast, Franz Kummert, *Member, IEEE*, Ute Ehrlich, Gernot A. Fink, Thomas Kuhn, Heinrich Niemann, and Gerhard Sagerer, *Member, IEEE*

***Abstract*—This article presents the speech understanding and dialog system EVAR. All levels of linguistic knowledge are used both to control the analysis process and for the interpretation of an utterance. All kinds of knowledge are integrated in a homogeneous knowledge base. The control algorithm used for the analysis is defined within the representation scheme and does not depend on the application.**

One of the aims of EVAR is to develop a system structure where linguistic and nonlinguistic expectations could be used not only for the interpretation but also as predictions for the recognition process.

***Index Terms*— Speech understanding, dialog system, dialog model, syntactic and semantic analysis, semantic network, problem-independent control.**

I. Motivation and System Overview

SPEECH is the preferred and natural means of communication for humans. This is a good reason for building systems that communicate with users via speech. An interesting domain for such speech understanding systems is information dialogs where the user wants to get some information by asking the system which takes the role of a "competent person" in the field of interest. In order to make such a communication process possible it is important that the system "understands" the utterances of the dialog partner and reacts to the understood information according to the expectations of the partner.

A speech recognition system will become a speech understanding system only if it incorporates a component for the interpretation of the meaning. Such an understanding component built for the *Speech Understanding and Dialog System EVAR* is described in this paper. For an overview of EVAR see [21], for a more precise description of the recognition component of EVAR, see [11].

Understanding requires an adequate representation of the meaning. This analysis in most systems is done after the recognition phase by finding an interpretation in the dialog context for the generated word chains. The linguistic levels, *syntax* (the structural relations between the words of an utterance), *semantics* (the interpretation of the meaning of an utterance), and *pragmatics* (finding truth values for the semantic interpretations in a concrete situation), are represented in most natural language (NL) systems for the analysis of written language and also in some speech understanding systems (see, for instance, [6], [8], [17], [33]) using representation techniques like the predicate calculus (e.g., [1], [10]), frames, or semantic networks (e.g., [7], [2], [31]).

At least for the recognition phase in systems for speech understanding statistical methods are used ([6], [15], [3], [18]). The disadvantage of these statistical methods is that they do not help to find a representation of the meaning of an utterance. They are adequate only to recognize the uttered sequence of words, using some linguistic knowledge to restrict the possible combinations of words to word chains. The resulting chains do not have to be grammatically correct even if they are very similar to the spoken utterance (e.g., differ only in one word or in one ending of a word). Therefore, the chain cannot necessarily be interpreted syntactically and semantically. For this reason in recent systems knowledge-based techniques are being used, either after the recognition process (e.g., [8], [33]) or to control the recognition process itself with context-based expectations (e.g., [17], [10]).

Only a few systems use the semantic features to control the analysis at the recognition level (e.g., [30], [8]). Here, such an approach is presented: all levels of linguistic knowledge can be used both to control the analysis process or for the interpretation of word chains. For this all the knowledge is integrated in a homogeneous knowledge base. The control algorithm used for the analysis is defined within the representation scheme. It does not depend on the application.

One of the aims in developing EVAR is to have a system structure where linguistic and nonlinguistic expectations could be used not only for the interpretation but also for the recognition process (see [4], [26]). This seems to be necessary because otherwise too many syntactic constituents can be found with the number of word hypotheses generated during the recognition process (see [14]). Words actually spoken don't need to be within these hypotheses and the interpretation could correct the wrong hypotheses or add the correct ones. This surely is not possible without a feedback to the speech signal. In [33], the conclusion is: "The only possible alternative I can see is to control the analysis from the very top of all the knowledge and search just for the events the system is interested in." This is taken into account in the system

Manuscript received May 20, 1991; revised Febreuary 19, 1993. Recommended for acceptance by Associate Editor R. DeMori.

M. Mast, T. Kuhn, and H. Niemann are with Lehrstuhl für Informatik 5, Mustererkennung, Universität Erlangen-Nürnberg, Martensstr. 3, 8520 Erlangen, Germany.

U. Ehrlich was with Lehrstuhl für Informatik 5, Mustererkennung, Universität Erlangen-Nürnberg, Martensstr. 3, 8520 Erlangen, Germany. She is now with Triumph-Adler AG, Olivetti Office Research, Fürther Str. 212, D-8500 Nürnberg 80, Germany.

F. Kummert, G. Fink, and G. Sagerer are with AG Angewandte Informatik, Technische Fakultät, Universität Bielefeld, 4800 Bielefeld, Germany.

IEEE Log Number 9214451.

architecture of EVAR: A semantic network is used with an integrated control algorithm in which high-level task-specific knowledge can be represented on different levels of abstraction. These levels reflect the syntax, semantics, pragmatics, and dialog knowledge. The strategy of the analysis is goal-directed using the acoustic evidence for hypotheses in an efficient way.

The application domain of EVAR is inquiries about German intercity train connections. The communication is to proceed via telephone and thus other channels of communication (e.g., visual) are excluded. This restriction to use speech implies a wide range of grammatical constructions in order to be able to represent all possible facts to be given to the dialog partner in a natural way. A special problem with spoken dialogs is that grammatical rules differ from those used in written text (e.g., see [9]): The structures of sentences are not as complex as in written text, but a lot of utterances are grammatically incomplete. Usually, after an initial utterance, dialogs consist (nearly) exclusively of sentence fragments which can only be interpreted—as it is also the case for anaphoric constituents like pronouns—within the dialog context. The analysis of utterances in a speech understanding system thus has to be controlled or at least supported by the context, i.e., by expectations about the possible structure and meaning of the actual user utterance. Spoken language dialog systems similar to ours were presented in [35], [36], [34], [16], [20].

The paper is organized as follows. In Section II, the linguistic knowledge used in EVAR is presented. Section III gives a brief description of knowledge representation in the semantic network system ERNEST as well as the representation of linguistic and dialog knowledge. How this knowledge is used in the analysis process is described in Section IV. Results and a short outlook will conclude the paper.

II. Linguistic Knowledge

For understanding a user utterance, the following levels of linguistic and domain dependent knowledge are distinguished.

Morpho-Syntactic Knowledge: It is used to search for and to build up simple syntactic constituents, i.e., syntactic units containing only one "nucleus" which can be the head of other words (e.g., "the next *train*"). The generation of complex constituents like "the next *train* | with *course* | to the *south*" is only done with additional semantic knowledge in order to prevent the generation of too many syntactically correct constituent hypotheses which are semantically inconsistent. Using only syntactic knowledge, the word "south" in the above example could depend on "course," "course" on "train," which is the semantically correct interpretation. However, it would also be possible to subordinate both "south" and "course" directly to "train," which semantically is not correct.

Syntactic-Semantic Knowledge: First the semantic knowledge is used to check the semantic consistency between the words of a word chain. Second the generation of longer word chains (i.e., complex constituents, see above, and whole sentences) is supported by both syntactic and semantic knowledge in order to use linguistic (i.e., here semantic) restrictions as early as possible.

Pragmatic Knowledge: In order to find an adequate answer to a user utterance it has to be interpreted within a special domain of application. In the system EVAR, this is the domain of intercity trains, departures, arrivals, prices, etc.

Dialog Knowledge: A user utterance has also to be interpreted within the situational context. That comprises both the knowledge of how to behave in the situation of an information request, what kind of utterance may follow each one, but also the consideration of the dialog history in order to be able to resolve references and to find the expected reaction.

In the following, we first give an overview of the knowledge needed for the analysis of one utterance. Following the contextual knowledge and the dialog model of the system are presented.

A. Analysis of an Utterance

1) Morpho-Syntactic Knowledge: A constituent grammar containing eight different types of constituents is used for the morpho-syntactic analysis. All these constituents are used in information requests. The constituent grammar does not comprise constructions that are used only for metacommunicative purposes like polite phrases or greetings. These are modeled in an additional "dialog grammar," which is directly referred to by the dialog module of the system (see Section I-C). Subordinate clauses, coordinations (with the exception of temporal adjuncts like "between 10 and 11 o'clock"), and negations are not considered so far. The constituents are the following.

[NG] noun group

- with a noun as nucleus: "*the/which/a big* suitcase"; the article and the adjective could be left out; numbers or ordinals can be added; chains of adjectives are possible, also with modifying adverbs, e.g., "a very big rather new suitcase"; there are no noun or prepositional groups dependent on the head noun (see above).
- with apposition: e.g., "the intercity train 'Deichgraf'/number 163" (only for trains).
- pronouns (reflexive, personal, or interrogative) or proper nouns are noun groups on their own (only without additions, i.e., not "the beautiful Hamburg").
- no coordinations (e.g., "Peter and John"); no comparisons ("as you") or adverbial modifications ("only you").

[PNG] prepositional group

- preposition with noun group: e.g., "on Tuesday", "during this weekend"; no postponed prepositions.
- preposition with adverb: e.g., "since today/when".

[ADJUG] predicative or adverbial adjective group: e.g., "(very/how) fast", "soonest"; no comparisons ("as fast as possible").

[ADVG] adverbial group: e.g., "when", "as always", "today".

[UHRZ] time of day: e.g., "between 10 and 12 o' clock", "five minutes to ten", "at what time".

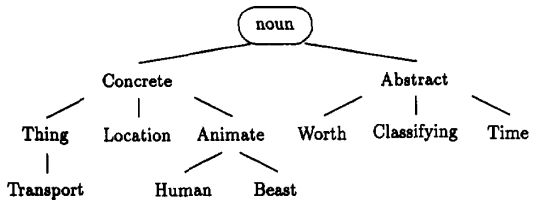


Fig. 1. Hierarchy of Semantic Classes for nouns (part).

[DATUM] date: e.g., "on Wednesday the 4th of Apr. 1990".
[INFG] infinitive group: e.g., "(he started) to work"; no words dependent on the infinitive; no passive constructions.
[VG] verbal group: e.g., "comes", "have written".

2) Semantic-Syntactic Knowledge: Semantic Consistency: To check the semantic compatibility between words of a word chain a semantic classification system is used where semantic classes are assigned to single words, e.g., "Location" to "Hamburg", or "Transport" to "train", or "Movement" and "Process" to "to leave". These classes are ordered hierarchically, for example, the class "Thing" comprises the class "Transport", i.e., the word "train" can represent also the class "Thing" (see Fig. 1). For some words (prepositions, adjectives) there exist selectional restrictions for the combinations possible with other words, especially nouns. The required compatibility is defined via the classification tree (see Fig. 1): If there is a connection from X to Y in the direction from the root to the leaves then Y is compatible with X, e.g., "Transport" is compatible with "Thing" and "Concrete", but not vice-versa. For example the word "fast" in the meaning of "a fast train" requires a noun that describes an object with the property that it can be moved or can move itself (e.g., of the class "Transport" but not of "Location" as the noun "town"). So the noun phrase "the fast town" has to be rejected as semantically inconsistent.

The checking of the selectional restrictions is also used to disambiguate different semantic (local) interpretations: e.g., the constituent "mit dem nächsten Zug" ("with the next train") in German represents $4 * 2 * 2 = 16$ different combinations of the possible semantic interpretations of the lexemes *mit*, *nächsten*, and *Zug*. This results from the number of meanings represented in our lexicon. But only one is semantically consistent (*mit* selecting a noun with the class "Thing", *nächster* selecting a noun with the class "Animate" or "Thing" or "Location", and the noun *Zug* with the class "Transport" or "Location"). All the other possible combinations do not have a common intersection of the noun's semantic class with the given selections of the preposition and the adjective. Since "Transport" is a specialization of "Thing" the meaning of the whole constituent can even be determined to be "Transport" because this is the only possible meaning fitting to all three words.

There are also other semantic features that can be used to check the semantic consistency of a word chain.

- Most nouns in German cannot be used with singular number but without article. To decide which singular noun does not need an article semantic knowledge is needed. Constituent hypotheses consisting only of a singular noun (noun groups) or of a singular noun with a preposition (prepositional groups) are acceptable only if the head noun is a mass noun with the semantic class "Continuous" (e.g., "water", "grass") or "Quantitative" (e.g., "with money"), or if it describes a profession, some function, the nationality (e.g., "teacher", "Dutch"), a property (e.g., "commodity", "speed"), a state (e.g., "illness"), or a process.
- There are also the additional semantic features TYPE and REFERENCE that are assigned to a constituent if it contains a word with special properties. For instance a constituent that contains the article "a" like "a train" has the attribute TYPE "indefinite", a constituent that contains the pronoun "my" like "my car" has the attribute TYPE "possessive", a constituent that contains a superlative adjective like "the earliest train" has the attribute TYPE "definite", or a constituent which contains a word referring to something in the actual situation like "here" or "my opinion" has the attribute REFERENCE "deictic". Not all the values of these features can be combined, for example the cardinal number "one" with the attribute TYPE "indefinite" cannot be used together with the superlative "next". So the constituent "one next train" is not acceptable semantically.

3) Semantic-Syntactic Knowlege—Complex Constituents and Sentences: The search for complex constituents and sentences is done using syntactic and semantic knowledge based on the valency theory (see e.g., [32]) and the case theory (see [5]).

The main idea is that the syntactic and semantic structures of a sentence are essentially determined by its head verb. The property to call for a certain number and kind of complementary noun groups or prepositional groups to build up an adequate sentence is called valency. The morpho-syntactic and semantic descriptions of the complements constitute a verb frame with slots (called actants) to be filled by actual phrases. For each expected phrase a functional role (a deep case) can be given. Since the caseframes differ from word to word, this information have to be contained in the lexicon of the system. The lexical knowledge base in EVAR provides caseframe entries for verbs but also for nouns and adjectives. Usually, alternative meanings correspond to different caseframes. A relatively detailed case system with about 30 domain independent cases is used (e.g., *Agent, Instrument, Cause*). Examples for caseframes are given in Fig. 2. For instance the caseframe "Verbindung.1.5" (connection) has two slots. Both are optional, i.e., they need not be realized. Both slots have to be filled with a constituent which has the syntactic type "prepositional group" where the semantic class of the noun has to be compatible with "Location". If the functional role of the constituent is *"Source"* then the semantic class of the preposition has to be compatible with "Origin", otherwise if the functional role is *"Goal"* then the semantic class of the preposition has to be compatible with "Direction".

In addition to these actants, which are defined by the head word of the constituent or the sentence, free adjuncts can be added nearly independently of the meaning of the head word. Currently only genitive constructions like "the dining car *of the train*" describing a part of a whole (deep case "Relation")

fahren.1.1 ("The train is *going* from Hamburg to Munich")
Instrument: noun group (nominative), Transport, obligatory
Source: prepositional group (Origin), Location, optional
Goal: prepositional group (Direction), Location, optional

fahren.1.2 ("I am *going* by train from Hamburg to Munich")
Agent: noun group (nominative), Animate, obligatory
Instrument: prepositional group (prep.="mit"), Transport, optional
Source: prepositional group (Origin), Location, optional
Goal: prepositional group (Direction), Location, optional

Abfahrt.1.1 ("the *departure* of the train at Hamburg for Munich")
Object: noun group (genitive), Transport, optional
Location: prepositional group (Place), Location, optional
Time: prepositional group (Moment), Time, optional

Verbindung.1.5 ("a *connection* from Hamburg to Munich")
Source: prepositional group (Origin), Location, optional
Goal: prepositional group (Direction), Location, optional

Fig. 2. Caseframes (examples).

or a possessive relation ("Possessive") and temporal adjuncts like "tomorrow morning" are considered.

The latter are very important for the application "information about intercity trains". Temporal constituents have to be handled in a special way because they can be chained together. The chaining results in new temporal constituents which have to be interpreted as a whole (for example "tomorrow | morning | at about nine o' clock"). The possible combinations of the single constituents are defined via a grammar reflecting the strict limitations given by morpho-syntactic, semantic, and pragmatic rules.

4) Pragmatic Knowledge: As an example of an application we use an information system which covers all the information about German intercity trains, for example information about the timetable, about fares, or about special services in intercity trains in general or of one special train. Seven different types of user questions are distinguished, and are ordered in a hierarchy (see Fig. 3). Each type of information can be described with the information needed to answer a special user request (for example the **destination**, i.e., the city to which the user wants to go). For more specialized types of questions this information are inherited from their ancestors in the tree of Fig. 3.

- **Information about trains:** This type represents the complete task domain.
- **Information about objects:** This is general knowledge about stations and trains, for example: "Do I need a supplementary ticket for intercity trains?".
- **Information about train connections:** This type covers all information about intercity train connections between cities. Here it is for example obligatory to specify the **destination**, i.e., where the user wants to go to. Another obligatory requirement is where the user wants to start. If this is not articulated it is assumed that the desired city for the departure is the city where the system is located.
- **Information about timetable:** Several trains might be running on a special route per day with the same destination and city of departure; this information is inherited by the **information about train connections**. In addition a time interval when the user wants to leave or when he wants to arrive has to be given.
- For each train the **information about special services** can be the focus of attention.
- Also the possibilities of reservation can be interesting for a user. So there is another type of question, **information about reservation**, which is valid for one train connection at a special time.
- For the **information about fare** the route and the possibility of a reduction are needed. Since this is not dependent on the timetable, this type of information is a specialization of the **information about train connections**.

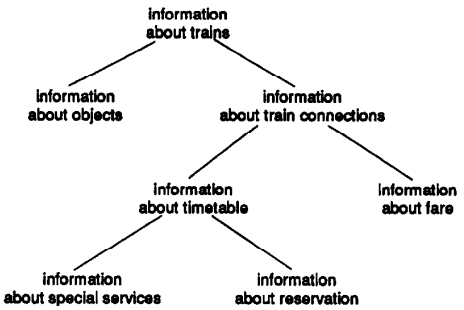


Fig. 3. Hierarchy of Types of Questions (the links are specialization links, where "information about train" is the most general concept)

B. Dialog Model

1) Interpretation in the Dialog Context: A special problem within a dialog situation where partners presuppose a certain amount of common contextual and situational knowledge is the determination of possible referential objects in the real world. This is done with the help of a dialog memory. The resolution of anaphorically used constituents, i.e., constituents referring back to some previously mentioned objects is of special interest. Currently several different linguistic possiblities to refer back are regarded. For the following examples it is assumed that they are preceeded by "You can take the intercity train at 8.30h":

1) When does *it* arrive in Hamburg? (personal reference)
2) Is there a dining car *in this train?* (definite - descriptive)
3) Is it possible to have breakfast *in the dining car?* (collocation).

Another important feature especially for speech is the frequent usage of elliptical constructions. Currently we concentrate on the analysis of ellipses which are generated using the linguistic constructions of the prior utterance. We distinguish two types of such ellipses which both are modeled by a special grammar for ellipses.

1) The "syntactic" ellipses, i.e., grammatically incomplete simple constituents where the head has to be taken from outside of the linguistic context, for example
Is it *the last (one)?* (nominal ellipsis; in German the "one" is not used).
2) The "semantic" ellipses, i.e., grammatically incomplete sentences where parts of the sentence like the verb or some of its actants are taken out of the linguistic context, for example

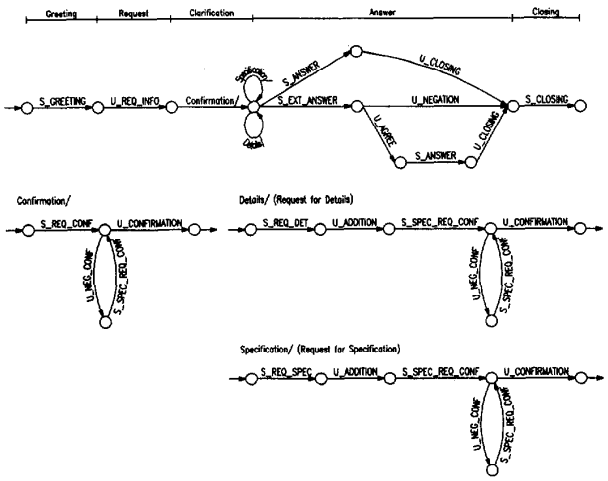


Fig. 4. The dialog model.

S: You can take the intercity train at 13.30h.
U: *13.30h.*

Both types can also be combined.

2) Dialog Model: As mentioned above, the user should have the possibility of talking to the system without too many restrictions, i.e., almost like talking to an information officer at the station. So the dialog model (see Fig. 4) must represent all dialog acts which are typical in this special situation. On the other hand, we achieved a simplification compared to real natural dialogs by guiding the user with special system utterances.

The dialog model so far contains the dialog initialization and ending phases and only one information request and answering cycle. If the information neccessary for giving an answer are not given in the user's request the system starts a clarification dialog (see Fig. 4). The user utterances have to be syntactically correct, i.e., they have to be syntactically and semantically complete or they have to be incomplete in a way such that they can be completed by taking parts of prior utterances (see Section III). In the following some examples for the different dialog phases are given.

greeting:
S: Hello. This is the Automatic Travel Information System. What information do you need?
request:
U: Tomorrow I want to go to Hamburg.
confirmation:
S: Have you said Hamburg?
U: Yes, Hamburg.
request for details:
S: Do you want to start at Nürnberg?
U: Yes, at Nürnberg.
S: Okay, tomorrow from Nürnberg to Hamburg.
U: Yes.
request for specification:
S: Do you want to leave in the morning or in the evening?
U: In the afternoon, but not too late.
S: Tomorrow early in the afternoon?
U: Yes, if possible.
answer:
S: You can take the train at 14.15h.
U: That's okay for me. Thank you very much.
closing:
S: You are welcome. Have a nice trip.

C. Knowledge about Answer Generation

The emphasis in the developed system is on the analysis of utterances in task-oriented dialogs in the domain of information provision services.

To enable the system to communicate in a spoken dialog with the user, and not only to answer questions like in a question-answer system a dialog component and also an answer generating component are needed.

For the answer generation answer schemes are used for each dialog act. Besides some metacommunicative acts, which control the phatic communication, dialog acts that are concerned with the domain are needed. The answer schemes for the latter acts need to be updated during the dialog.

Example: Request for confirmation of destination and time of arrival.

In the answer scheme for requests for confirmation:
"Sie wollen in ORT ZEIT ankommen."
("You want to arrive at PLACE TIME.")
the variables for destination ORT (PLACE) and arrival time ZEIT (TIME) have to be replaced by the actual parameters to produce the following output:
"Sie wollen in München am 4. Juli zwischen 18 und 21 Uhr ankommen."
("You want to arrive at Munich on the 4th of July between 6 and 9 p.m.")

Apart from times and places the result of the database request, e.g., a connection, has to be filled in an answer scheme.

1) Database Access: To enable the system to answer requests in the domain of train timetables and prices, database access is needed. For this reason an intercity knowledge module for the German Intercity net was developed which provides connections and prices corresponding to the parameters given by the user. Input needed for the database request are the parameters the user gives about the connection he needs. These are at least the destination and an interval for the departure or arrival time. For the departure place the system uses a default (the city in which the system is located), if nothing else was uttered. If one obligatory parameter is missing, the dialog module has to start a request for it.

Before the database retrieval, several consistency checks are performed, e.g., the given time interval should not exceed a certain limit, otherwise the set of retrieved connections will be too large. Then the database is searched for all suitable connections which match the given parameters. Therefore all intercities with all stops and departure and arrival times must be available. Out of the retrieved connections the best ones are collected, that means the ones with a minimum of changes and a minimum of detour. For each of these connections the

intercity-trains and the departure, change, and arrival times and places are available.

2) Dialog History: Most references in an utterance refer to the last utterance. In the case of user utterances the last system utterance is relevant. For the resolution of all references the whole dialog must be available. All dialog steps including the system utterances have to be stored in the dialog history.

III. Representation in a Homogeneous System

We briefly describe a framework for the representation of declarative and procedural knowledge based on a suitable definition of a semantic network. Apart from the framework for knowledge representation the system includes a control strategy which is problem independent (see Section II-B). A complete software system, called ERNEST, has been implemented in Section II-C for this purpose.

A. Formalism

Besides the declarative part of a knowledge base which can model objects, events, and other problem specific knowledge, procedural knowledge which gives information on how the declarative knowledge can be used for the interpretation of patterns is needed. In the following, the syntax, semantics and pragmatics of the available data structures are described. Further details are described in [26], [22]. With this knowledge representation language it is possible to model a certain section of the real world and a certain aspect of this section.

1) Nodes: In our definition of a semantic network three types of nodes are distinguished. The nodes model concepts, classes of concepts or modified concepts, which allow the representation of constraints resulting from actual data, or are descriptions of individuals.

Concept: Represents classes of objects, events, or abstract conceptions, for example, syntactic constituents, deep cases or verb frames.

Instance: A subset of the sensor data which can be associated with a certain concept, for example an interval of the signal which can be associated with a concept for a certain syntactic class.

Modified Concept: A concept which is constrained by the already available instances in an intermediate state of processing, for example during the analysis a modified concept for the syntactic class "article" can be generated with restrictions regarding gender, case and number if it is regarded as part of a NOUN_GROUP and the nucleus is already instantiated (which restricts the gender, case and number for the other parts in this constituent).

2) Links: Links are used to express relations between the nodes. Apart from the links to instances, three different types of links and with them three organizational axes are distinguished, which define a partial order on the set of concepts.

Specialization: Refinement of a more general concept which inherits the properties like part, concrete, attribute, and so on unless something else is mentioned (these properties can be modified or deleted).

Part: A concept may consist of certain parts. This relation between a concept and its parts is represented by a part link. For example a NOUN_GROUP can consist of one article and one noun (this is not the only possibility). It often occurs that a certain part can only be recognized in the context of the corresponding object having this part. For example a certain deep case obtains its meaning only in the context of a caseframe. Therefore, concepts for deep cases are defined as *context-dependent* parts of verb frames.

Concrete: With concrete links, concepts of different conceptual systems can be connected, while part and specialization relationships are only within the same conceptual system. For analysis purposes conceptual systems must be ordered in a hierarchy of levels of abstraction. For example, for the linguistic knowledge four conceptual systems - syntax, semantics, pragmatics, and dialog - can be used.

The data structures of the three node types are identical. The nodes are described by attributes, relations, and judgements which are necessary for the analysis process.

A concept may have obligatory and optional parts and/or concretes. A *modality set* is the set of obligatory parts and concretes together with the associated set of optional parts and concretes sufficient to instantiate a concept. One concept can be defined by several modality descriptions. e.g., one modality description of the concept NOUN_GROUP has the obligatory parts ARTICLE and NOUN and the optional parts ADJECTIVE, NUMBER, ORDINAL NUMBER, and NEGATION. For each modality description a temporal or spacial order on parts and concretes can be defined in an *adjacency description.*

3) Attributes: For a physical object or an event certain attributes, for example number, gender, case or duration are usually needed.

Furthermore, analysis parameters which are required only for a more efficient analysis can be defined, for example an analysis parameter "semantic class" is useful in certain concepts of the pragmatic level, with regard to restrictions of the semantic class out of pragmatic facts.

The main items of the attribute description are "role", "type of value" and "computation of value". "Role" means the functional role of the attribute. The item "computation of value" contains a function which computes an actual value of the attribute given by the sensor data. The "judgement" is a computation of a score for the attribute.

For example an attribute with the role "gender" can be defined in the concept "NOUN_GROUP" on the syntactic level. "Type of values" is a set with a maximum of three members, which are "masculine", "feminine", and "neutral" for German. The computation of value has to determine the gender for the NOUN_GROUP from the gender of the parts. Therefore, the attribute gender of the parts is argument for the computation of value of the attribute gender in the concept NOUN_GROUP.

4) Relations: Certain relationships between parts and/or concretes of a concept can be defined in a structural relation. e.g., the attributes gender, case, and number of the parts of the concept NOUN_GROUP must agree. The relation description contains among others a "role" and a "judgement" which is a function testing the relation.

5) Judgement: The item judgement of a concept contains a function computing a "judgement" of an instance or a modified concept. Arguments to this function are the judgement of the links, attributes, and relations. The judgement is a tuple of different scores (see Section II-A).

6) The Pragmatics of the Formalism: Another important aspect is the utilization of this network for a dialog system. Given certain sensor data the main activity is to compute instances out of concepts.

The instantiation process is defined by the following rules that are the basis for the problem-independent control. The rules are defined for the whole network without respect to the task domain.

RULE 1 says that in order to compute an instance of a concept "A" there must be instances of all its concretes and parts which are obligatory for some modality set. Requiring an instance of a part is only possible if it is a context independent part, e.g., the concept GOAL (which represents a deep case) is a context dependent part of the concepts NF_INTERCITY and VF_REISEN (which represent the noun frame "intercity train" and the verb frame "to travel"). For the instantiation of the concept GOAL there must be at least an instance of either NF_INTERCITY or VF_REISEN. A problem could exist in computing an instance of the concept VF_REISEN which has the context dependent part GOAL, because for the instantiation of VF_REISEN, an instance of GOAL is needed while for the instantiation of GOAL an instance of VF_REISEN is needed. This problem is solved in RULE 1 by computing a *partial instance* of VF_REISEN requiring only instances of context independent parts.

Having a partial instance of VF_REISEN an instance for GOAL can be computed and with this instance for GOAL the partial instance of VF_REISEN can be completed with RULE 2.

RULE 3 checks whether there are instances of optional parts or concretes. In this case an *extended instance* is created by adding these parts. e.g., an instance of the concept NOUN_GROUP having instances for the concept ARTICLE and the concept NOUN which are obligatory in a modality set, can be extended by an instance of the concept ADJECTIVE which is an optional part of the same modality set.

Given a goal concept for an analysis process, recursive application of these three rules results in a search tree for the goal concept.

If some instances have been computed but instantiation of the concept "A" is not yet possible, it may be possible to compute a modified concept of "A". RULE 4 describes the data driven creation of modified concepts, e.g., a modified concept of the concept NOUN_GROUP can be created if for the concept ARTICLE a new modified concept or instance was created. With this rule a bottom-up restriction, e.g., of attribute values in the modified concept NOUN_GROUP is possible. e.g., if the article for the attribute gender has the value "feminine", in the modified concept of NOUN_GROUP the attribute gender can be restricted to "feminine" too.

RULE 5 summarizes a model driven creation of modified concepts. With the inverse computations of values which are associated with the corresponding computations of values in attributes, relations, and links, top-down restrictions can be made. In the above mentioned example from the modified concept of NOUN_GROUP a modified concept of the concept NOUN (which is referred by a part link) can be created. By an inverse computation of value the attribute gender can be restricted to "feminine" too. Thus RULE 4 and RULE 5 provide the bottom-up and top-down propagation of constraints in the network.

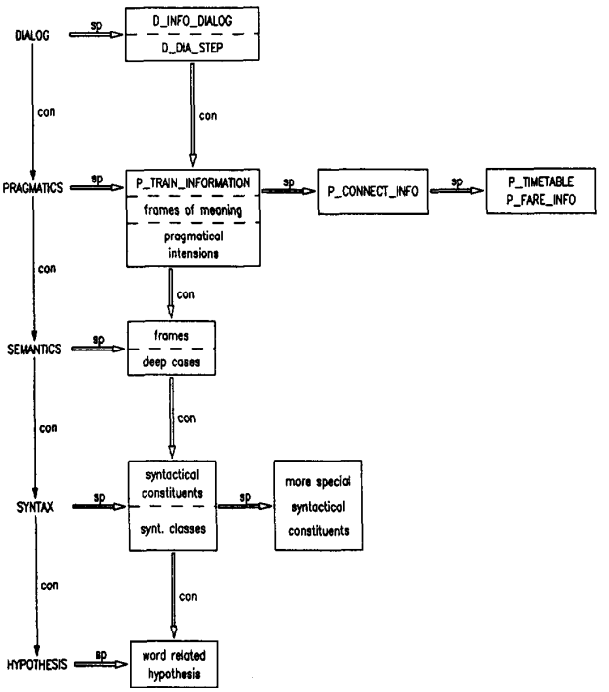


Fig. 5. Overview of the network for the interpretation of inquiries about German intercity train connections. Each block in this figure stands for a collection of concepts having an identical level of abstraction and an identical depth in the specialization hierarchy. The blocks are connected by specialization (sp) links and concrete (con) links. Inside a block, concepts are connected via part links.

B. Semantic Network Representation of the Linguistic Knowledge

For the representation of the linguistic knowledge a homogeneous hierarchical knowledge base using the above described system ERNEST was created (see [28]). An overview is given in Fig. 5.

It represents the syntax and semantics of a subset of the German language, knowledge about the task domain "intercity-train-information" as well as dialog knowledge. Therefore, four conceptual systems for the linguistic knowledge base were created. On the lowest level of abstraction (see Fig. 5) the concepts for word related hypotheses build an interface between the linguistic analysis and the word recognition.

1) Syntactic Knowledge: On the syntactic level, syntactic classes and larger syntactical units are modeled (see Section II-A-1). Each concept for a syntactic class has a concrete link to a concept for a word hypothesis on the lowest level

of abstraction. For the description of syntactic classes the attributes gender, number, case, semantic class, pragmatic class, and metacommunication are defined which correspond to the slots of the lexicon entries.

Larger syntactical units are the constituents which are built up by the syntactic classes. For example concepts for noun phrases (SY_NG) or times (SY_UHRZ) are modeled. A simple noun group has part links to the concepts for noun, pronoun, interrogative pronoun, relative pronoun, and proper name. The optional and obligatory parts are defined in the modality description. The time sequence of these parts is defined in the adjacency description. Concepts include the attributes gender, number, and case ensuring the syntactic correctness as well as the analysis parameters semantic and pragmatic class which ensure semantic and pragmatic compatibility (see Section II-A-2). Especially in speech, special forms are used for the metacommunicative parts of a dialog e.g., for greetings and thanks. For these utterances special syntactical units were modeled.

2) Semantic Knowledge: The semantics is based on the valency theory and the case theory (see Section II-A-3). The semantic level contains concepts for deep cases and verb and noun frames. There exist concepts for 13 different deep cases, which are connected with the syntactic level by concrete links. They provide additional syntactic restrictions, for example, a preposition list which can be used in the prepositional phrase connected with a special deep case.

24 verb and 37 noun frames are represented, for example the verb frame "arrive": by the concept S_VF_ANKOMMEN. Each of them is connected by a part link with the deep cases the frame opens. For each meaning of a verb or noun a modality description exists which defines the obligatory and optional deep cases.

3) Pragmatic Knowledge: The next linguistic level reflects the pragmatics given by the task domain "intercity-train-information" (see Section II-A-4). Actually concepts are modeled for

- the different pragmatic goal concepts P_CONNECT_INFO and P_TIMETABLE;
- frames of meaning e.g., P_VF_FAHREN (which contain restrictions resulting from the meaning in the actual application); and
- pragmatic intentions e.g., P_DESTINATION.

Frames of meaning with the possible pragmatic intentions as parts are analogous to the frames and deep cases on the semantic level. They provide additional restrictions resulting from the task specific usage.

The pragmatic goal concepts model the different topics the system can deal with and they contain the pragmatic intentions the system is able to talk about as part links. e.g., the P_TIMETABLE has the parts: P_DESTINATION, P_DEP_PLACE, P_TO_TIME, P_FROM_TIME and so on.

A network detail is shown in Fig. 6.

4) Dialog Knowledge: On the level which represents knowledge about dialog, a dialog in the domain of information provision services is modeled as well as the dialog acts which it consists of (see Section II-B-2). Actually a simplified dialog

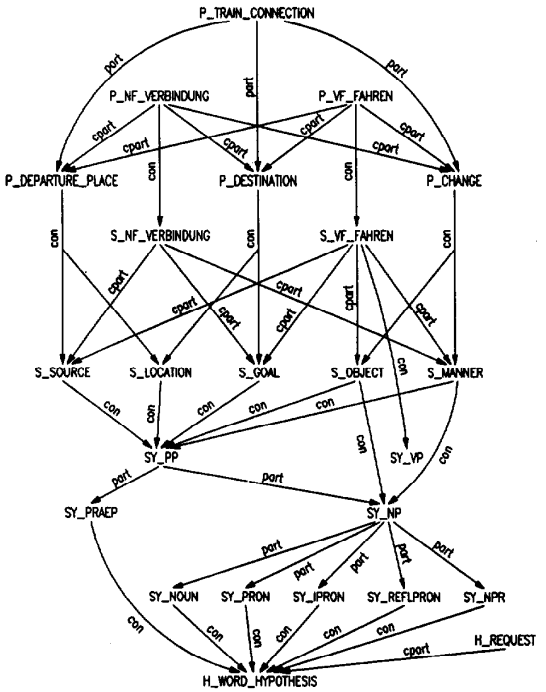


Fig. 6. Detail of the network for speech understanding covering the levels of words, syntax, semantics, and pragmatics. The concepts are connected by concrete links (con), part links (part) and context-dependent part links (cpart).

is foreseen which will allow to test the described parts in one homogenous environment and therefore gives the chance to first communications with the system.

The dialog is represented by a sequence of dialog acts which can be metacommunicative or concerned with the application. Each dialog act is modeled by a concept with concrete links to the syntactic, semantic, and pragmatic levels which provide information about the corresponding realization. Dialog acts with metacommunicative functions can be represented by syntactic or semantic units whereas those which are concerned with the application are represented by pragmatic units.

Further information about the dialog acts provide the attributes metacommunication, intonation, and word order which contribute to deciding which dialog act is realized by the actual data.

5) Interface to Acoustics: On the lowest level of abstraction, concepts are modeled which build an interface between word recognition and the linguistic analysis. They gather all available restrictions (e.g., case nominative, gender masculine, number singular, semantic_class Transport) during the analysis process and thereby constrain the possible instances for a concept. The instances of these concepts are computed from the actual set of word hypotheses.

Input for the linguistic analysis process is a set of word hypotheses. A hypothesis is a quatruple (w, a, e, s) where w denotes the hypothesized word. s denotes the acoustic score, and $[a, e]$ specifies the time interval which is covered by the hypothesis. During the analysis process a second interface to acoustics enables the verification of word chains (see Section

II-A). If needed a part of the speech signal is given to the verifier with a chain of words. Then the verifier computes a score for the word chain in matching it with this part of the speech signal. The generation of word hypotheses as well as the verification is based on Hidden Markov Models (see [11], [29]).

E. Semantic Network Representation of the Intercity Data Model

The IC-knowledge was integrated in the described knowledge base. Therefore, the network environment was expanded by concepts which describe certain connections, IC train stations, IC departure times as well as price information.

After the retrieval of connections out of the data base (see Section II-C-1), for each connection a concept is generated which contains the trains which participate as well as the change places and times. All connections given by a set of parameters are finally represented by a general concept which refers to them. That means that during a dialog only information about the actually needed connections is available. This is an effective reduction of data.

IV. Analysis Strategy

In Section III-A-6, the inference rules for knowledge utilization were presented. Their recursive application builds up the skeleton of the search space for the analysis strategy. Competing word or word chain hypotheses together with competing linguistic results split up this skeleton into the complete search space. The search in this space is directed by the A*-Algorithm [23]. In the next section, we explain the different scoring values used for the control of the analysis. Then, we give an outline of the analysis strategy illustrated by an example.

A. Judgement

For a goal directed search the currently most promising hypothesis should be selected for further processing. Therefore, results from different levels of analysis (word recognition, syntax, semantics, pragmatics, dialog) should be comparable to reach this goal. For an adequate description, the judgements should reflect terms like the following.

- **compatibility**: Is the hypothesis contradictory to the model?
- **quality**: Measure for the correspondence of signal/model.
- **reliability**: How likely is it that a hypothesis is correct?
- **relevance**: What is the priority of the hypothesis for further processing?

To enforce a more model driven strategy neither a left to right nor an island driven strategy is used. On every location in the speech signal a word hypothesis is accepted if it is in accordance with the expectations from the linguistic model. Only when two word hypotheses are adjacent with respect to the speech signal, is a word chain built and verified by the acoustic module. Therefore, a hypothesis H in the context of linguistic processing is a collection of word and word chain hypotheses with a linguistic interpretation. Each of these hypotheses, represented in our system by a search tree node, has a judgement vector with the following components. Full details of the judgements are given in [21], [28], [27].

- f
- **Structural compatibility**: This is a binary measure, which tests the linguistic restrictions, i.e., congruence of case, number, and gender in a noun group. That means:

$$z(H) = \begin{cases} 1, & \text{if all restrictions are fulfilled by } H, \\ 0, & \text{otherwise.} \end{cases}$$

- **Acoustic quality** of the underlying word or word chain hypotheses + **estimate** for the not covered speech signal: The acoustic score is generated by the EVAR word verification module and is the negative logarithmic probability of a continuous density Hidden Markov Model [29]. To guarantee the comparability of short and long hypotheses a statistical optimistic estimate for the acoustic quality of the unmasked speech signal is calculated, which is based on the distribution of correct hypotheses. It has been shown in [27] that mean and variance of the quality q_k of correct hypotheses depend linearly on the length L of a hypothesis, hence

$$\begin{aligned} \mu_k(L) &= E\{q_k | L, \text{correct}\} = \mu_k L \\ \sigma_k^2(L) &= E\{(\mu_k(L) - q_k)^2 | L, \text{correct}\} = \sigma_k^2 L. \end{aligned} \quad (1)$$

With these formulas a statistically optimistic estimate for the not covered speech signal of length L is given by:

$$\tilde{q}(L) = \mu_k L - C\sigma_k\sqrt{L}. \quad (2)$$

That means, the acoustic quality for the not covered speech signal is estimated by the mean value of correct hypotheses ($\hat{=}\mu_k L$). For an optimistic estimation, C-times of the standard deviation ($\hat{=}\sigma_k\sqrt{L}$) is subtracted. If the acoustic quality (resulting from the underlying word or word chains) of a hypothesis H is given by $q(H)$, then

$$\hat{q}(H) = q(H) + \tilde{q}(L) \quad (3)$$

is a comparable measure for the acoustic score of a hypothesis H. More details are given in [28].

- **Number of frames of the word chain with longest duration**: As the quality of word hypotheses reflects a distance-measure the following statement is valid for hypotheses with equal quality: hypotheses with longer duration are more probably correct hypotheses than shorter ones [27]. Therefore,

$$\begin{aligned} s(H) &= \max_{1 \le i \le N} \{L(K_i)\}, \\ L(K_i) &:= \text{length of chain } K_i \end{aligned} \quad (4)$$

is a measure for the reliability.

- **Number of masked frames**: Measure of relevance, because the analysis goal can be reached in fewer steps:

$$r(H) = \text{number of masked frames for } H. \quad (5)$$

Written in a vector, the judgement $b(H)$ of a hypothesis H is

$$b(H) = (z(H), \hat{q}(H), s(H), r(H)). \tag{6}$$

The comparison between two hypotheses is defined by the lexical order of their judgement vectors, i.e.,

$$(x_1, \cdots, x_4) < (y_1, \cdots, y_4) \Leftrightarrow \exists x_i [x_i < y_i], \\ 1 \leq i \leq 4 \wedge \forall x_l [x_l = y_l], l < i.$$

This means first x_1 and y_1 (structural compatibility) are compared. If they are equal, then x_2 and y_2 (acoustic score) are compared and so on. This is done until one component is greater than the corresponding one in the other vector. Moreover, for the second and third components of the vectors, only interval values and not the exact values are used.

B. Control

The goal of the analysis of an utterance is the instantiation of a concept representing a type of user question (see Section II-A-4). These concepts reflect the possible requests and contain all the information needed for a database request. Due to the uncertainty of the word generation module, a strictly data driven analysis does not seem to be too promising. Above all, the syntactic restrictions are insufficient to avoid an excessive expansion of the search tree. Analogously, a strictly model driven strategy was not successful because speech offers a lot of possibilities to express a certain fact. Therefore, we use a strategy which works both on the acoustic data as well as on the expectations from the linguistic model.

1) Initial Phase: In the following, the analysis process is demonstrated by the example: "Ich möchte nach München fahren." ("I want to go to Munich".). The analysis starts with a data driven generation of word hypotheses. Out of this set the n best judged and pragmatically relevant word hypotheses (e.g., München [Munich], Sonntag [Sunday]) will be selected as starting points for further processing. This is justified by the fact that pragmatically relevant words are pronounced with more emphasis and ensure therefore a better detection in the speech signal [24]. Experiments with spoken utterances showed that $n = 10$ is an appropriate value.

For every such hypothesis an instance of the corresponding syntactic class is created due to RULE 1 and 2 (see Section III-A-6). As every path from the start node to a node in the search tree represents a consistent partial interpretation, a search tree node is generated for every instance and is inserted as a competing successor of the start node. The judgement vector of these nodes is calculated from the corresponding instance and the related word hypothesis as described in the last section. Fig. 7 shows the complete search tree after that initial phase. In the following $I_k(X)$ stands for the k-th instance of the concept X and $Q_i(X)$ for the i-th modified concept to (X).

2) Estimation of Pragmatic Intensions: To use the powerful constraints of the pragmatic level the instances of the initial phase will be connected with appropriate pragmatic intentions (see Section III-B-3). The word hypothesis "München" [Munich] can be interpreted as "departure place" or as "destination", but not as "from time". On the contrary, for the hypothesis "Sonntag" [Sunday] "from time" respectively "to time" are

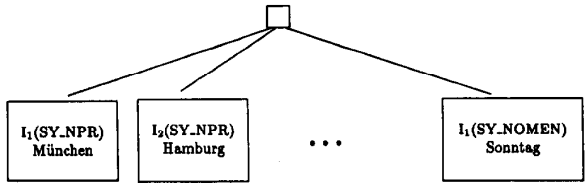


Fig. 7. Search tree after the initial phase ($I_k(X)$ stands for the k-th instance of the concept X).

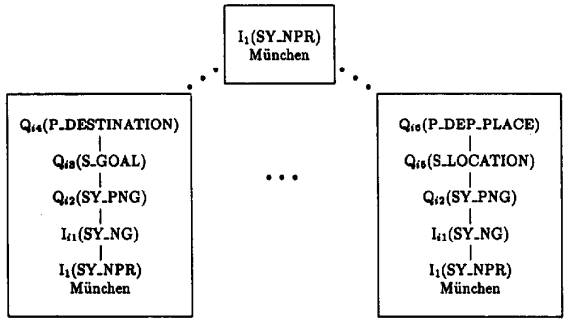


Fig. 8. Contents of search tree nodes after the estimation of pragmatic intentions.

adequate associations. To guarantee a correct association, the pragmatic intentions contain the attribute "pragmatic class". For a concrete pragmatic intention only certain values are allowed, i.e., "destination" $\overset{pragm_class}{\longrightarrow}$ "*town-with-an-intercity-station*". For every pragmatically relevant word the proper pragmatic classes are inserted in the lexicon. In consideration of that attribute possible paths in the network are constructed beginning from the initial instances up to an appropriate pragmatic intention. This is done by an iterative application of RULE 1, 2 or 4. Fig. 8 shows the contents of competing search tree nodes resulting from the initial hypothesis "München" [Munich]. They represent partial linguistic interpretations as "to go to Munich" and "to go from Munich". For a clear representation and an efficient processing all the information created from the starting node to a node n is collected in node n. The index of the instances and modified concepts represents the sequence in which these objects were created.

3) Syntactic Verification of the Pragmatic Intensions: In the next step, due to the expectations of a pragmatic intention the syntactic constituent will be completed. In the case of our example, the concept P_DESTINATION restricts the possible prepositions to "in" [in] and "nach" [to]. This is propagated by the iterative applications of RULE 5 to S_GOAL, SY_PNG, and SY_PRAEP. Additionally, the admissible areas on the time axis can be restricted for the preposition. Due to the adjacency matrix in the concept SY_PNG the preposition has to be located directly before the hypothesis "München" [Munich]. Therefore, the word recognition can be constrained exactly by the model driven information. By application of RULE 1 and 2 instances to the concepts SY_PRAEP and SY_PNG are created. The contents of a search tree node after that phase are shown in Fig. 9.

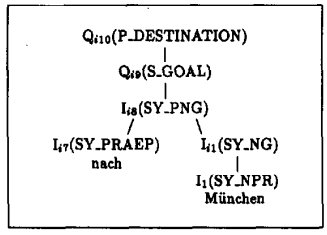


Fig. 9. Contents of search tree nodes after the syntactic verification of a pragmatic intention.

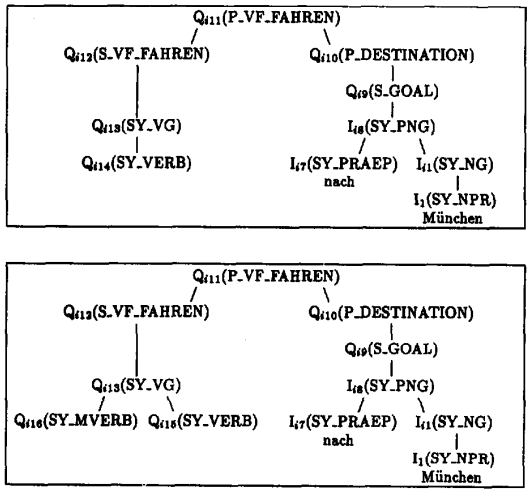


Fig. 10. Contents of two search tree nodes with an expanded model of a verb frame.

4) Verification of an Appropriate Context: As the pragmatic intentions are context dependent on verb frames or noun frames an appropriate context is needed for the instantiation. In the slot context-of of a pragmatic intention all possible contexts are referred to. For the concept P_DESTINATION among other things the frames "fahren" [to go], "Zug" [train], or "Verbindung" [connection] are admissible. For every context a modified concept is created by RULE 4 and inserted in a search tree node. By iterative application of RULE 5 the linguistic model is expanded and the necessary hypotheses can be requested. Fig. 10 shows two search tree nodes with a fully expanded model of the verb frame "fahren" [to go]. The lower one represents a verbal group with a modal verb and the upper one without a modal verb.

For the lower node a hypothesis for the verb "fahren" [to go] is requested with the following constraints:

- tense: infinitive
- restricted area of the speech signal due to the hypotheses "nach" and "München".

By RULE 1 and 2 the verbal group "möchte fahren" [want to go] is instantiated and a partial instance of S_VF_FAHREN is created. Thereby, S_GOAL can be instantiated too (see Fig. 11).

5) Estimation of Information Concepts: If a context is established an appropriate information concept is estimated. For the concept P_DESTINATION the concepts P_CONNECT_INFO

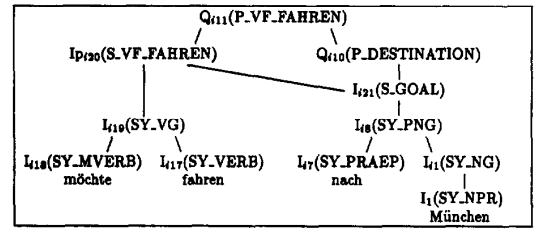


Fig. 11. Contents of a search tree node after the verification of a context.

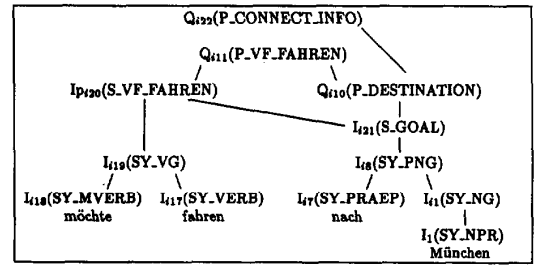


Fig. 12. Contents of a search tree node after the estimation of an information concept.

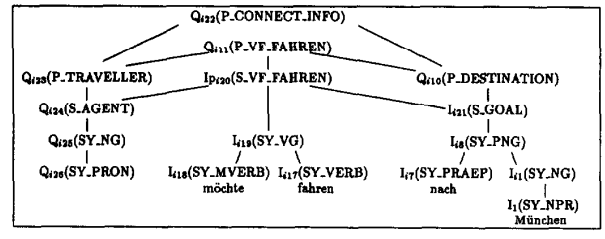


Fig. 13. Contents of a search tree node during the verification of an information concept.

and P_TIMETABLE are admissible which would result in two competing search tree nodes. Since P_TIMETABLE is a specialization of P_CONNECT_INFO all information generated for P_CONNECT_INFO can be used by P_TIMETABLE. Therefore, only the most general appropriate information concept is estimated (see Fig. 12). If an instance of P_CONNECT_INFO is not sufficient to interpret the whole speech signal, the so far generated instances can be used to instantiate P_TIMETABLE. Otherwise, these instances have to be created twice on different search tree paths.

6) Verification of Information Concepts: In this phase the control alters between an expansion of the model by RULE 5 (see Fig. 13) and an instantiation of the model by RULE 1, 2, 3 until the information concept is instantiated (see Fig. 14). In this case, the analysis of the utterance terminates when the speech signal is interpreted sufficiently.

In the other case, one tries to instantiate optional links or special concepts. In our example P_TIMETABLE is an adequate specialization of P_CONNECT_INFO and is inserted as a new goal concept for the analysis. For a timetable information a departure time or a destination time is obligatory. Due to the frame "fahren" [to go] only the concept P_FROM_TIME is

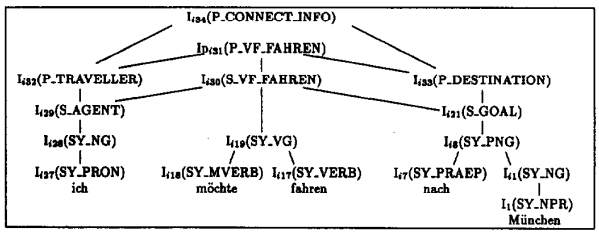


Fig. 14. Contents of a search tree node with an instantiated information concept.

appropriate. The process of expansion, instantiation, and determination of new goals is repeated until the above condition for termination is fulfilled.

During the first three phases of the analysis the search tree is fully expanded to guarantee linguistically motivated partial interpretations for the further processing. After that, the A*-Algorithm with the judgement vector of Section IV-A is used to direct the analysis. For further details see [12].

V. Results and Outlook

Before we present results obtained with our system, the experimental framework is described.

1) The basis for the experiments are continuously spoken dialogs sampled with 16kHz.
2) In detail, in the knowledge base the following concepts are realized:
   a) A concept for the information dialog D_INFO_DIALOG and 7 concepts for the dialog acts, e.g., "user information request" and "system answer"
   b) Two information concepts P_CONNECT_INFO, and P_TIMETABLE as well as the proper pragmatical intentions (10 concepts).
   c) The frames for 37 nouns and 24 verbs as well as the proper deep cases (13 concepts).
   d) All syntactic constituents presented in Section IV-A except the infinitive group (7 concepts).
3) The analysis is directed by the judgement vector described in section 2.1.
4) The lexicon used for word recognition and verification contains 1081 inflected forms.
5) The linguistic coverage comprises single sentences. The sentences can be elliptical provided they can be completed by parts of prior utterances.

Dialog evaluation is still a much debated research topic, so that some remarks with respect to our evaluation of the system are given. For the evaluation of dialog systems no generally used measures or tests are available. Furthermore the following points make it more difficult to evaluate a dialog system.

- First a dialog system cannot be tested in batch mode with input from a file because the system reaction has to be taken into consideration for the next input. This means no predefined dialog corpus can be used, rather each dialog has to be tested by a human which takes a lot of time.
- Because there are few existing German spoken language dialog systems and fewer dialog systems in the domain of train time tables, no test corpus like the ATIS database is available.

In order to test the system not only with sentences from the system developers, we exchanged test corpora with people working on similar systems [19], [25] in the ESPRIT-Project Sundial, however these test corpora contain only single utterances and the rest of the utterances in each dialog were formed according to the reaction of the system.

To judge the efficiency of the complete system the following two groups of experiments were executed, in the following they are denoted *Test1* and *Test2*.

Test1: Speaker-dependent version of the acoustic module realized at the University of Erlangen.

- Analysis with up to 100 word hypotheses (depending on the duration of the utterance) per dialog act uttered by the user.
- The experiments run on a DEC RISC station 5000 with 32MB main memory and 25 mips.
- The word recognition and the verification modules are speaker-dependent.
- The acoustic module was trained with 100 domain specific and 200 phonetically balanced sentences.
- During the generation of word hypotheses a bigram model of perplexity 111 is used.
- For a user request 90 % of the speech signal has to be covered by word hypotheses.
- A dialog can consist of up to 5 dialog acts. The user starts with a request for information after an optional greeting. Then the system either asks back for a missing parameter (which is needed to start a database request) or asks for confirmation. The user gives the missing parameter, a confirmation, or a correction. These dialog acts can consist of one single word, an elliptical construction, or a complete sentence. Finally, the system starts a database request for a suitable connection, fills an answer pattern with this information, and generates an answer with a speech synthesizer.
- 85 dialogs were tested. 50 user requests were taken from a German corpus of 100 sentences (e.g., user requests, confirmations) which was created for the ESPRIT-Project Sundial. 35 user requests were taken from a test corpus created for the EVAR system. These user requests were tested in a natural language mode and after the system reaction a suitable answer (e.g., confirmation) was given. This was done in order to have reference dialogs which can be tested in the spoken language mode. In the following an example dialog (translated into English) is given:

  user: when can I go to Munich tomorrow morning
  system: you want to go to Munich tomorrow morning
  user: yes to Munich
  system: *output of the appropriate trains*

  The 85 dialogs in total consist of 350 dialog acts (system plus user) and 873 words (user only).

- The experiments were executed as follows. For each dialog the test speaker read the first sentence of the dialog and the analysis process began. The second user utterance depends on the system reaction. If the user request was analysed correctly (that means the system reaction was the same as in the natural language mode), the second user utterance tested in the natural language mode was spoken. Otherwise the user utterance was adjusted to the system reaction. In the following example the time reference wasn't analysed correctly and the system asked back for it. However in the natural language mode the user request was analysed correctly therefore no asking back was necessary. In this case, the test speaker tried and suceeded to complete the dialog successfully by giving an appropriate answer (see the following example).

	natural language mode	spoken language mode
user:	when is the next train to Hamburg	when is the next train to Hamburg
system:	you want to take the next train to Hamburg	when do you want to go to Hamburg
user:	yes	now
system:	*output of the appropriate trains*	*output of the appropriate trains.*

  In the case of failure (e.g., total failure of the analysis), the recording was repeated.
- The generation of the word hypotheses after recording each utterance takes 3.53 times realtime (without special hardware). The word accuracy was 90.86 %, word correct 91.08 % and sentence correct 54.21 %.

The results are presented in Fig. 15. 85 dialogs with 170 user utterances (one utterance can consist of more than one dialog act) were tested. 68 times the dialog was completed succesfully. In three of them, a successful completion was possible after the system failed to analyze the user request (e.g., one pragmatical intention was missing) but the user corrected the system after the request for confirmation. 17 dialogs were not completed successfully, which means that the data base retrieval did not provide the connections which the user asked for, because of an incorrect analysis or a failure due to space limitations.

At the sentence level 129 of 170 utterances were interpreted correctly. That means the result of the analysis was the correct (corresponding) dialog act based on a correct syntactic, semantic and if needed pragmatic analysis. Additionally, 15 utterances were instantiated with the correct dialog act and information concept but with an incorrect or missing expression for the time or another pragmatic intention. 7 utterances were not interpreted correctly. For 19 utterances the analysis failed.

The average time to complete a dialog was 3:57 minutes. The average CPU time for the linguistic analysis to complete a dialog was 1:32 minutes. On the average 1390 search tree nodes and 24 MB space were needed for the completion of a dialog. On the average an utterance was repeated 1.14 times before an analysis was possible.

number of dialogs	85
successfully completed dialogs	68 (80%)
dialogs completed (with clarification)	3
average time to complete a dialog	3:57 min
average CPU time to complete a dialog	1:32 min
average number of search tree nodes	1390
failure of the analysis due to space limitation	11 (13%)
failure due to an incorrect analysis	6 (7%)
number of utterances	170
correct analysis	129 (76%)
incomplete pragmatical intention	15 (9%)
false pragmatical intention	7 (11%)
failure of analysis	19 (4%)

Fig. 15. Summary of the *Test1* results.

Most of the time in the *Test1* experiments the information given in the first user request was analysed correctly by the system and therefore the dialogs could be completed within only two dialog steps of the user.

Test2: Multi-speaker system (4 male speakers), realized at the University of Bielefeld.

- For the word recognition and verification task the ISADORA-System [29] is used.
- The acoustic module works without a language model. (This means the perplexity is about the lexicon size)
- The acoustic module is trained with 500 domain specific sentences from every speaker.
- The linguistic analysis works with the word hypotheses resulting from the 10 best-scored word chains of the acoustic module.
- For each dialog the test speaker speaks the first sentence of the dialog into the microphone and the analysis process begins. Due to the interpretation of the system a suitable answer (confirmation, correction, additional information) is given. This process is repeated until the dialog is successfully completed or failed.
- The tests run on a DEC RISC station 5000 with 32MB main memory and 25 mips.

Since the acoustic module works with high perplexity and in 4-speaker mode a word accuracy of only 74.6% was achieved. Therefore, in many sentences not all of the spoken words are hypothesized for the linguistic analysis. To manage this problem the requirements for the coverage of the speech signal with word hypotheses are reduced and an extended version of the dialog [13] compared to *Test1* is integrated into the system. For a successful linguistic analysis the minimal coverage of the speech signal with word hypotheses is set to 2/3 (see Section II–B). This allows in many cases a correct interpretation of the utterance inspite of missing spoken words. Together with a dialog strategy which requests information missing for a data base inquiry by a check-back partial interpretations can be completed. Furthermore, incorrect interpretations can be corrected by a clarification dialog. During that dialog phase the user can confirm or correct all parts of the interpretation until the desired information is available.

One speaker of the training phase tested the system by 50 dialogs. The results are presented in Fig. 16. 60% of

the dialogs were completed successfully without an incorrect interpretation. On the average 2.7 user utterances were made per dialog. The following dialog (translated into English) gives a typical example. Since the coverage of the speech signal was 2/3 only the words in bold font were used for the linguistic interpretation.

user: **hello** in the morning **I want to go to Munich**

system: when do you want to go to Munich

user: **in the morning**

system: You want to go from Bielefeld (Default) to Munich in the morning

user: **yes**

system: *output of the appropriate trains*

14% of the dialogs could be completed successfully after a clarification phase. For these dialogs 3.4 user utterances had to be done on the average. The following dialog is a typical example.

user: **is there a train** from Munich **to Hamburg** next sunday **at seven** (spoken was eleven) **o'clock**

system: You want to go from Bielefeld (Default) to Hamburg at seven o'clock

user: **no from Munich** next sunday **at eleven o'clock**

system: You want to go from Munich to Hamburg at eleven o'clock

user: **no next sunday at eleven o'clock**

system: You want to go from Munich to Hamburg next sunday at eleven o'clock

user: **yes that's okay**

system: *output of the appropriate trains*

On the average, for one utterance of the completed dialogs the linguistic analysis consumed 25.5 seconds of CPU-time and built up 478 search tree nodes.

For the remaining 26% no successful dialog was carried out. The main reason was the insufficient acoustic analysis hypothesizing less spoken words than needed for the coverage. Therefore, with an improved word recognition module the results also can be improved.

At the sentence level 45 of 106 utterances were interpreted correctly. That means the result of the analysis was the correct (corresponding) dialog act based on a correct syntactic, semantic, and pragmatic analysis. Additionally, 42 utterances were instantiated with the correct dialog act and information concept but with a missing or incomplete pragmatic intention. 8 utterances were not interpreted correctly and for 11 utterances the analysis failed.

The results in Fig. 16 show that the integration of a flexible dialog strategy allows the successful treatment of problems like missing spoken words or incorrect interpretations.

Further improvements will extend the linguistic competence of the system. This includes the use of prosodic information to support the linguistic analysis, the improvement of the resolution of anaphoric references, the interpretation of utterances containing more than one sentence, and the modeling of spontaneous speech phenomena.

successfully completed dialogs (without corrections)	60%
successfully completed dialogs (with clarification)	14%
no successful dialog	26%
number of utterances	106
correct analysis	45 (42.5%)
missing or incomplete pragmatical intention	42 (39.6%)
incorrect interpretation	8 (7.5%)
failure of analysis	11 (10.4%)

Fig. 16. Summary of the *Test2* results.

REFERENCES

[1] H. C. Bunt, "On-line interpretation in speech understanding and dialogue systems," in *Recent Advances in Speech Understanding and Dialog Systems*, H. Niemann, M. Lang, and G. Sagerer, Eds. Berlin: Springer-Verlag (NATO ASI Series F), 1988, vol. 46, pp. 349–396.

[2] A. Cappelli, G. Ferrari, L. Moretti, and I. Prodanof, "A framework for integrating syntax and semantics," in *Computational Models of Natural Language Processing*, B. G. Bara and G. Guida, Eds. Amsterdam: Elsevier (North-Holland), 1984, pp. 33–57.

[3] P. D'Orta, M. Ferretti, A. Martelli, and S. Scarci, "An automatic speech recognition system for the italian language," in *3rd Conf. European Chapter of the ACL*, Copenhagen, 1987, pp. 80–83.

[4] U. Ehrlich, "Bedeutungsanalyse in einem sprachverstehenden System unter Berücksichtigung pragmatischer Faktoren," *Sprache und Information*, vol. 22. Tübingen: Max Niemeyer Verlag, 1990.

[5] J. C. Fillmore, "A case for case," in *Universals in Linguistic Theory*, E. Bach and R. T. Harms, Eds. New York: Holt, Rinehart and Winston, 1968, pp. 1–88.

[6] P. K. Fink, "The acquisition and use of dialogue expectation in speech recognition," UMI Dissertation Information Service, Ann Arbor, MI, 1983.

[7] W. S. Havens, "Recognition mechanisms for schema-based knowledge representation," in *Computational Linguistics*, N. J. Cercone, Ed. Oxford: Pergamon, 1983, pp. 185–200.

[8] P. J. Hayes, A. G. Hauptmann, J. G. Carbonell, and M. Tomita, "Parsing spoken language: A semantic caseframe approach," in *Proc. 11th COLING*, Bonn, 1986, pp. 587–592.

[9] L. Hitzenberger and H. Kritzenberger, "Simulation experiments and prototyping of user interfaces in a multimedial environment of an information system," in *Eurospeech 89: European Conf. Speech Commun. Technol.*, Eurospeech Congress, Paris, Sept. 1989, pp. 597–600.

[10] J. P. Ingria, "Natural language processing: Where it's been and where it might be going" in *Computer Processing of Language Data (ROJP). Proc. 4th Conf.*, Portoroz, YU, 1988, pp. 59–74.

[11] T. Kuhn, H. Niemann, E.G Schukat-Talamazzini, W. Eckert, and S. Rieck, "Context-dependent modeling in a two-stage hmm word recognizer for continuous speech," in *Signal Processing VI: Theories and Applications (EUSIPCO-92)*, J. Vandewalle and A Oosterlinck, Eds. Amsterdam: Elsevier, 1992, pp. 439–442.

[12] F. Kummert, "Flexible Steuerung eines sprachverstehenden Systems mit homogener Wissensbasis," *Dissertationen zur Künstlichen Intelligenz*, vol. 12, Infix, Sankt, Aug. 1992.

[13] F. Kummert, G. Fink, G. Sagerer, and B. Seestaedt, "Erweiterungen einer dialogkomponente zur robusten verarbeitung gesprochener sprache," Interner Bericht, AG Angewandte Informatik, Universität Bielefeld, 1992.

[14] S. Kunzmann, T. Kuhn, and H. Niemann, "An experimental environment for generating word hypotheses in continuous speech," in *Recent Advances in Speech Understanding and Dialog Syst ems* (NATO ASI Series F), H. Niemann, M. Lang, and G. Sagerer, Eds. Berlin: Springer Verlag, vol. 46, 1988, pp. 311–316.

[15] S. E. Levinson and L. R. Rabiner, "A task-oriented conversational mode speech understanding system," *Bibliotheca Phonetica*, vol. 12, pp. 149–196, 1985.

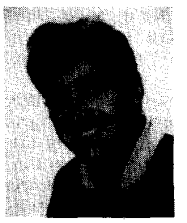
[16] L. M. Norton *et al.*, "Management and evaluation of interactive dialog in the air travel domain," in *Proc. DARPA Workshop*, June 1990, pp. 141–146.

[17] J. Mudler and E. Paulus, "Expectation-based speech recognition," in *Recent Advances in Speech Understanding and Dialog Systems*, (NATO ASI Series F), H. Niemann, M. Lang, and G. Sagerer, Eds. Berlin: Springer-Verlag, vol. 46, 1988, pp. 473–477.

[18] H. Ney, D. Mergel, A. Noll, and A. Paeseler, "Overview of speech recognition in the spicos system," in *Recent Advances in Speech Understanding and Dialog Systems*, (NATO ASI Series F), H. Niemann, M. Lang, and G. Sagerer, Eds. Berlin: Springer-Verlag, vol. 46, 1988, pp. 305–310.
[19] G. Niedermair, "Lingustic modeling in the context of oral dialogue," in *Int. Conf. Spoken Language Processing*, Banff, AB, Canada, Oct. 12–16, 1992, pp. 635–638.
[20] ——, "Syntax, semantik und dialog in SPICOS II," in *Sprachliche Mensch-Maschine-Kommunikation*, H. Mangold, Ed. München: Oldenbourg-Verlag, 1992, pp. 91–102.
[21] H. Niemann, G. Sagerer, U. Ehrlich, G. Schukat-Talamazzini, and F. Kummert, "The interaction of word recognition and linguistic processing in speech understanding," in *Speech Recognition and Understanding* (NATO ASI Series F 75), P. Laface and R. DeMori, Eds. Berlin, Heidelberg: Springer-Verlag, 1992, pp. 425–453.
[22] H. Niemann, G. Sagerer, S. Schröder, and F. Kummert, "Ernest: A semantic network system for pattern understanding," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, pp. 883–905, Dec. 1990.
[23] N. J. Nilsson, *Principles of Artificial Intelligence*. Berlin: Springer-Verlag, 1982.
[24] E. Nöth and R. Kompe, "Der Einsatz prosodischer information im spracherkennungssystem evar," in *Mustererkennung 88, 10. DAGM-Symposium Zürich, Informatik-Fachberichte*, H. Bunke, O. Kübler, and P. Stucki, Eds. Berlin: Springer-Verlag, 1988, pp. 2–9..
[25] N. Youd P. Heisterkamp, S. McGlashan, "Dialogue semantics for an oral dialogue system," in *Int. Conf. Spoken Language Processing*, Banff, AB, Canada, Oct. 12–16, 1992, pp. 643–646.
[26] G. Sagerer, *Automatisches Verstehen gesprochener Sprache, Reihe Informatik*, vol. 74. Mannheim: Bibliographisches Institut, 1990.
[27] G. Sagerer, U. Ehrlich, F. Kummert, H. Niemann, and E. G. Schukat-Talamazzini, "A flexible control strategy with multilevel judgements for a knowledge based speech understanding system," in *Proc. 9th Int. Conf. Pattern Recognition*, Rome, 1988, pp. 788–790.
[28] G. Sagerer and F. Kummert, "Knowledge based systems for speech understanding," in *Recent Advances in Speech Understanding and Dialog Systems* (NATO ASI Series F), H. Niemann, M. Lang, and G. Sagerer, Eds. Berlin: Springer-Verlag, 1988, vol. 46., pp. 421–458.
[29] E. G. Schukat-Talamazzini, H. Niemann, W. Eckert, T. Kuhn, and S. Rieck, "Acoustic modeling of subword units in the isadora speech recognizer," in *Proc. Int. Conf. Acoust., Speech and Signal Processing*, San Francisco, CA, vol. 1, pp. 577–580, 1992.
[30] M. Shigenaga, Y. Sekiguchi, T. Yagisawa, and K. Kato, "A speech recognition system of continuously spoken japanese sentences and an application to a speech input device," in *Proc. ICASSP*, Tokyo, 1986, pp. 1577–1580.
[31] N. K. Sondheimer, R. M. Weischedel, and R. J. Bobrow, "Semantic interpretation using KL-ONE," in *Proc. 10th COLING*, Prague, Czechoslovakia. 1984, pp. 101–107.
[32] L. Tesniere, *Elements de syntaxe structurale*, 2nd ed. Paris: Klincksieck, 1966.
[33] G. Thurmair, "Semantic processing in speech understanding," in *Recent Advances in Speech Understanding and Dialog Systems*, H. Niemann, M. Lang, and G. Sagerer, Eds. Berlin: Springer-Verlag (NATO ASI Series F), vol. 46, 1988, pp. 397–420.
[34] V. Zue, J. Glass *et al.*, "Recent progress on the Voyager system," in *DARPA Speech and Natural Language Workshop*, June 24–27, 1990.
[35] S. J. Young and C. E. Proctor, "The design and implementation of dialogue control in voice operated database inquiry systems," *Comput. Speech & Language*, vol. 3, no. 4, pp. 329–353, 1989.
[36] S. R. Young, A. G. Hauptmann, W. H. Ward, E. T. Smith, and P. Werner, "High level knowledge sources in usable speech recognition systems," *Commun. ACM*, vol. 32, pp. 183–194, 1989.

Marion Mast obtained the diploma in computer science from the University Erlangen-Nürnberg, Germany, in 1988.

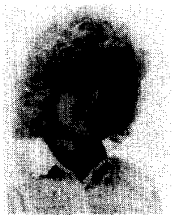
Since 1989, she is with the speech group of the Institute for Pattern Recognition at the University of Erlangen-Nürnberg. Her current research interests are speech understanding especially on dialog level and dialog systems.



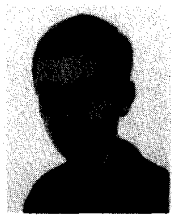
Franz Kummert (M'91) received the diploma and the Ph.D. (Dr.-Ing.) degree in computer science from the University of Erlangen-Nürnberg, Erlangen, Germany, in 1987 and 1991, respectively.

From 1987 to 1990, he worked at the "Institut für Informatik, Mustererkennung" (pattern recognition), at the University of Erlangen-Nürnberg, Erlangen. Since 1991, he has been with the group "Angewandte Informatik" (applied computer science) at the University of Bielefeld, Germany.

His fields of research are speech and image understanding. His main interest is the control of knowledge-based signal understanding systems. He has published various papers in these fields, and is author of a book on the control of a speech understanding system.



Ute Ehrlich was born in 1959 in Meisenheim/Glan, Germany. She received a diploma degree in computer science (diplom Informatiker) and Dr. Ing. from the University Erlangen-Nürnberg in 1984 and 1989, respectively. From 1984 to 1990, she was with the Institute for Computer Science (Pattern Recognition), University Erlangen-Nürnberg, working in the field of development of the lexicon component and the semantic and prag matic analysis for the Automatic Speech Recognition and Dialogue System EVAR (co ordinated project by the federal Ministry for Research and Technology BMFT, ESPRIT project SUNDIAL). Since 1990, she has been a Systems Analyst at TA Triumph-Adler AG, Project Manager for Handwriting Recognition and Project Manager for ESPRIT projects EuroCoOp and EuroCODE) (both in the field of Computer Supported Cooperative Work) and for ESPRIT project TWB II (Translator's Workbench).

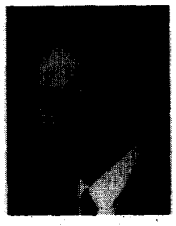


Gernot A. Fink received the diploma in computer science from the University of Erlangen-Nürnberg, Germany, in February 1991.

Since March 1991, he is working at the University of Bielefeld, Germany, with the group "Angewandte Informatik" (applied computer science). His field of research is speech recognition and understanding. His main interest is the interaction between word recognition and linguistic interpretation.

Thomas Kuhn was born in Hagen, Germany, in 1962. He received the diploma degree in computer science from the University Erlangen-Nürnberg, Erlangen, Germany, in 1989.

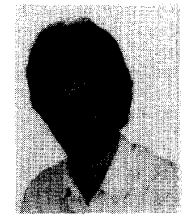
Since July 1989, he is with the speech group of the Institute of Pattern Recognition at the University Erlangen-Nürnberg. His current research interests are speech recognition and language modeling. Furthermore, he works on the interaction between the recognition and understanding level of a dialogue system.



Heinrich Niemann obtained the degree of Dipl.-Ing. in electrical engineering and Dr.- Ing. at Technical University Hannover in 1966 and 1969, respectively.

From 1967 to 1972, he was with Fraunhofer Institut für Informationsverarbeitung in Technik und Biologie, Karlsruhe, working in the field of pattern recognition and biological cybernetics. During 1973–1975, he was teaching at Fachhochschule Giessen in the Department of Electrical Engineering. Since 1975, he had been Professor of Computer Science at the University of Erlangen- Nürnberg, since 1988, he is also head of the research group "Knowledge Processing" at the Bavarian Research Institute for Knowledge Based Systems (FORWISS) and he is on the board of directors of the Institute.

His fields of research are image and speech understanding and the application of artificial intelligence techniques in these fields. He is on the editorial board of *Signal Processing*, *Pattern Recognition Letters*, *Pattern in Recognition and Image Analysis*, and the *Journal of Computing and Information Technology*. He is the author, coauthor, or editor of nine books and about 140 technical articles. He is a member of ESCA, EURASIP, GI, and VDE.



Gerhard Sagerer (M'88) received the diploma and the Dr.-Ing. degree in computer science from the University of Erlangen-Nürnberg, Erlangen, Germany, in 1980 and 1985, respectively. In 1990, he received the venia legendi (Habiliation) in computer science from the Technical Faculty of this University.

From 1980 to 1990, he was with the Institut für Informatik, Mustererkennung (pattern recognition) at the University of Erlangen-Nürnberg. Since 1990, he has been a Professor of Computer Science at the University Bielefeld, Germany. He is presently member of the academic senat of this University and vice dean of the Technical University. He is on the Scientific Board of the Forum InformatikerInnen für Frieden und gesellschaftliche Verantwortung, FIFF (German Section of Computer Scientists for Peace and Social Responsibility). His fields of research are image and speech understanding including artificial intelligence techniques and the application of pattern understanding methods to natural science domains.

Dr. Sagerer has published more than 50 technical papers in the areas of speech understanding and knowledge based image understanding with applications in medicine and industrial scenes, is author of two books and coeditor of another book on knowledge representation for image understanding and on the architecture of speech dialog systems. Dr. Sagerer is member of the German Computer Society (GI) and the European Society for Signal Processing (EURASIP).