Plan Recognition in verbmobil

Jan Alexandersson

DFKI GmbH

Report 81
Mai 1995

Mai 1995

Jan Alexandersson

DFKI GmbH
Stuhlsatzenhausweg 3
66123 Saarbrücken

Tel.: (0681) 302 - 5347
Fax: (0681) 302 - 5341
e-mail: `alexandersson@dfki.uni-sb.de`

Die vorliegende Arbeit wurde im Rahmen des Verbundvorhabens Verbmobil vom Bundesministerium für Bildung, Wissenschaft, Forschung und Technologie (BMBF) unter dem Förderkennzeichen 01 IV 101 K / 1 gefördert. Die Verantwortung für den Inhalt dieser Arbeit liegt bei dem Autor.

Plan Recognition in VERBMOBIL

Jan Alexandersson*
DFKI GmbH,
Stuhlsatzenhausweg 3
66123 Saarbrücken, Germany
{alexandersson}@dfki.uni-sb.de

Abstract

VERBMOBIL, a speech to speech translation system, poses new and challenging tasks for a dialogue component. We present a part of the dialogue component of VERBMOBIL: a robust plan recognizer that processes all of the over 200 dialogues of the VERBMOBIL corpus. The recognizer currently handles unexpected sequences of input, and will in the future deal with multiple input, hypotheses, and incomplete input.

1 Introduction

The speech to speech translation project VERBMOBIL [6] combines the two key technologies of speech processing and machine translation. In our scenario two dialogue participants, a German and a Japanese businessman, use English as common language for negotiating a meeting. They are, however, able to use VERBMOBIL to translate parts of the negotiation dialogue from their own native languages to English.

The processing environment of the VERBMOBIL system differs from other state-of-the-art dialogue systems in a number of ways, of which the following two have important consequences for the tasks and the design of a dialogue component.

- Instead of being a dialogue partner itself[1]. (i.e. controlling the dialogue) VERBMOBIL mediates the dialogue. This means that we have to assume that every utterance is a legal dialogue step, even if it is not consistent with our dialogue model. We thus have to be flexible enough to cope with *any* input irrespective of our expectations.

*This paper also appeared in the Proceedings of the IJCAI-95 Workshop "The Next Generation of Plan Recognition Systems: Challenges for and Insight from Related Areas of AI". Thanks to Norbert Reithinger, Elisabeth Maier, and Elizabeth Hinkelman for valuable comments and suggestions on earlier drafts of this paper.

[1]Unless VERBMOBIL is performing a clarification dialogue.

- Only the translated utterances are processed in depth (the rest of the dialogue is only shallowly processed) while a normal system analyses all the utterances in depth. We thus have to cope with unreliable data, probabilities and even gaps in our input.

In VERBMOBIL the dialogue component has the following tasks:

- to follow the dialogue and support other components with contextual information about the dialogue,
- to provide the analysis side with top down predictions for limiting search space.

In this paper we will focus on one of the parts of the dialogue component in VERBMOBIL, namely the plan recognizer. Faced with problems and requirements like real time, robustness, vague input, we show how to design such a component utilizing both knowledge based methods as well as statistical methods.

The outline of this paper is as follows: In section 2 we briefly present the VERBMOBIL system and the design of our hybrid component. Section 3 present our dialogue corpus which we have used to both infer the dialogue model and train our components. Special requirements are presented in section 4, and in section 5 we present our approach for plan recognition and show how to use statistical methods for improving run time performance. The paper concludes with section 6 where open questions and future work are presented.

2 An overview of VERBMOBIL

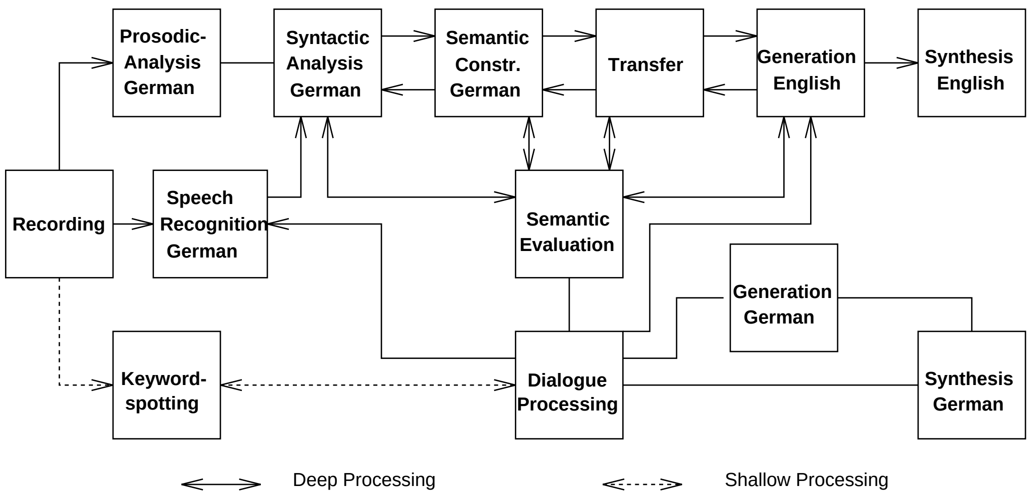

Figure 1: The architecture of the first VERBMOBIL demonstrator

The task of VERBMOBIL is to follow the dialogue and to translate an utterance on demand. By pressing a button the user causes VERBMOBIL to change to translation mode.

From the dialogue component's point of view, the VERBMOBIL system (see fig. 1) provides two main sources of input, namely the output from linguistic analysis (*deep processing*) and the output from the keyword spotter (*shallow processing*). Besides a speech recognizer, the linguistic analysis consists of a prosodic analyzer, a syntax parser, a semantic construction/evaluation component, a transfer, and finally a generation component. The keyword spotter is a speech recognizer trained on small data sets of different vocabularies with words typical for a certain dialogue step.

Like previous approaches to task oriented dialogue, we use a dialogue act based approach. We assume that our dialogues can be modeled by a limited but open set of dialogue acts. To process the dialogue, a multi-leveled hybrid approach has been chosen [1]. In this paper we concentrate on:

- **The statistical component** The task of this component is to predict the dialogue act(s) to come. It uses statistic information collected from a corpus of dialogues. Besides supporting the analysis with predictions, this information is used to guide the plan recognizer (for instance, when faced with unexpected input).

- **The plan recognizer** The plan recognizer serves two main purposes. We use it for building an intentional structure of the dialogue, a structure used, for instance, to support transfer and generation. It is also used for building the dialogue history. The plan recognizer has to be robust, fast and able to cope with unexpected input.

In VERBMOBIL the computation of the dialogue acts is done by the semantic evaluation component together with the dialogue component (in case of deep processing) and by the dialogue component on the basis of the output from the keyword spotter (in case of shallow processing). We give our top down predictions both to linguistic analysis, for limiting the search space, and to the keyword spotter, for suggesting which word set(s) to use. On the basis of the output from the keyword spotter we compute the likelihood of each dialogue act[4].

3 The VERBMOBIL Dialogues

As mentioned above, dialogue acts form the basis for dialogue processing. The idea is that an utterance will yield a better translation if the intentional aspect of the utterance is taken into consideration. In VERBMOBIL currently 54 domain dependent and domain independent dialogue acts are defined [3] (see fig 2). At present we use 17 of these (see fig 3) which are (with a few exceptions) domain independent[2]. We have analyzed over 200 transcribed dialogues and annotated them with these dialogue acts. An analysis of the corpus shows that three main dialogue phases can be distinguished.

[2]Investigating a travel planning scenario it was found that the current set (with small adjustments) also fulfills the requirements of such a scenario.

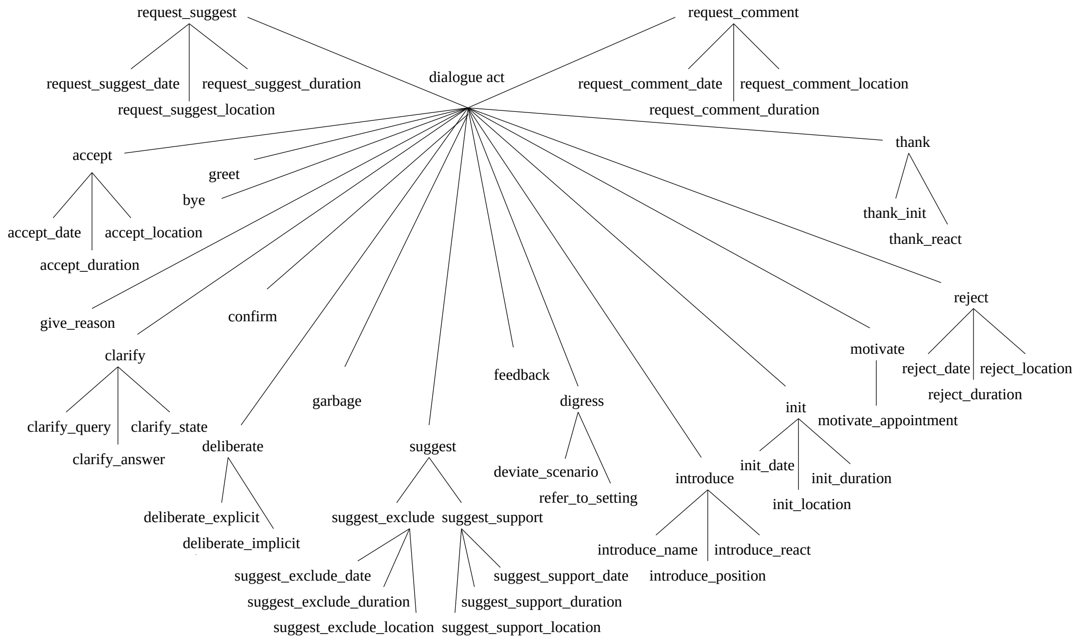

Figure 2: The dialogue act hierarchy

A typical dialogue begins with an introduction or greeting phase where the dialogue participants introduce themselves and the topic (for instance that one appointment should be scheduled) is verbalized. Then a negotiation phase follows where the actual negotiation takes place, and finally the dialogue proceeds into the closing phase where the appointment is confirmed and the dialogue participants thank each other before closing the dialogue. For each dialogue phase there are some typical dialogue acts. For the opening phase we have for instance GREET and INTRODUCE, for the negotiation phase SUGGEST, ACCEPT and REJECT, and for the closing phase CONFIRM and BYE. We have defined a dialogue model (see fig 3) which covers approximately 90% of the dialogues. Problematic are the dialogue acts in the subnet under the main net. These can appear anywhere in the dialogues, and have to be taken care of separately.

4 Requirements

In contrast to most other Natural Language systems, VERBMOBIL is not a dialogue partner itself. This is a big disadvantage since VERBMOBIL and in particular the dialogue component does not *control* the dialogue, and thus has no impact on how the dialogue proceeds. Processing a dialogue the dialogue component has to cope with a number of problems:

- **Deviations** Our dialogues contain dialogue acts (see figure 3) which can appear anywhere in the dialogues. For this reason we process them with a special repair mechanism (described in section 5.3).

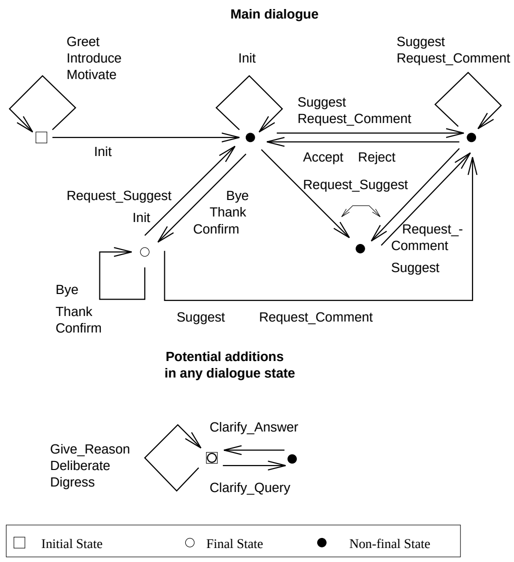

Figure 3: The dialogue model

- **Reinterpretations** Some utterances get different interpretations depending on the context (`Am Dienstag kann ich nicht!` (Tuesday is bad for me!) could be interpreted as a negative suggestion on a new appointment, or an rejection of a previous suggested one). The linguistic analysis might suggest the "wrong" dialogue act. If the suggestion does not fit into the dialogue model we will have to re-interpret the utterance.

- **Multiple Readings** An utterance might also represent more than one dialogue act. This is a highly contextual phenomena, where, for instance, an utterance like `Ich kann am Dienstag (Tuesday is possible for me)` can represent both a rejection (of an earlier suggested meeting) and a suggestion (of a new one), or just a suggestion (of a new meeting), or an acceptance (of a previously proposed one).

- **Real time constraints and Robustness** VERBMOBIL is aimed to be a robust real time machine. Our part (as well as other parts of the system) has to be implemented using efficient algorithms, which do not fail whatever input we are faced with.

5 The Plan Recognizer

5.1 The Basic Algorithm

We have designed a plan operator language, in which we currently have specified about 60 plan operators for building up the intentional structure. The underlying machinery is based on the relation between plan recognition and parsing as pointed out by Vilain [5].

The author shows that plan recognition in some cases can be viewed as parsing. He also shows an example how to compile a plan hierarchy into a context free grammar. This connection is convenient since we can use well known parsing strategies for recognizing a plan. Vilain then argues that a "chart-based version of Early's algorithm" can be used, an algorithm which we cannot adopt straight forward. Our plan operators contain side effects for instance for building the dialogue memory. We can therefore allow for only one structure to be "active[3]". Moreover, the Early algorithm is a *recognizer* which means that it either accepts or rejects the input. We are not allowed to reject the input and have to use different repair strategies when the input is not covered by our grammar. In the first version of our plan recognizer a simple top down algorithm with backtracking, (mimicking the behavior of a Prolog interpreter) has been used.

5.2 Guiding the search

To guide the search we use both manually added, and compiled out constraints, and a corpus based statistic method.

- **Constraints** The plan operator language allows for the programmer to add constraints. The constraints mostly address the context but can also be used to check pragmatic constraints e.g. if the dialogue participants know each other or not. To prune the search space even more we compile out so-called "reachable constraints". These constraints prevent a plan operator to be applied if there is no possibility for its successful application at a certain stage of the planning process. In the current version of our system these have been manually added to the constraints as mentioned above.

- **A Corpus-based Statistical Method** Sometimes the plan recognizer has more than one alternative operator to choose from. Instead of following a left-to-right search strategy (or some other fixed strategy), we collect statistics from a corpus of dialogues. This statistic is later used to suggest the most probable branch.

The idea is simple: Let the plan recognizer run over a corpus of dialogues, constructing *every* parse tree (plan) for each dialogue. For each time a certain plan operator successfully has been used to build the parse tree it gets one point. However, the use of the scores has to be constrained. Otherwise a plan operator can (mis-) use scores achieved in a certain context for being preferred in another context, as illustrated by the following example:

Consider the part of a grammar (with scores) in figure 4. Now, consider the case

$$\begin{array}{llll} (i) & a & \rightarrow & b\ 15 \mid d\ 0 \\ (ii) & a' & \rightarrow & b\ 5 \mid c\ 17 \end{array}$$

Figure 4: Example grammar with scores

when the plan recognizer has to choose between b or c in rule (ii). If the context in

[3] It is of course possible to keep multiple contexts in parallel, but for efficiency reasons we prefer to keep just one instead and when necessary reset the side effects when changing branch.

which the points has been achieved is not considered, rule b will be preferred since it has a total score of 20 (15 + 5), which is more than the 17 that rule c has. This is not correct since in rule (*ii*) it is more likely that c succeeds than b. We thus has to remember in what *context* the scores was achieved. Note also that it is important to construct *every* parse tree for each input and not just the first. If we just construct the first parse tree, we can not say that one plan operator is to be preferred before another, since we do not know whether this is true or not.

Calistri [2] uses a similar technique for choosing the most probable branch. He however *manually* adds the probabilities to different plan operators. Our approach allows us to dynamically change the scores to adjust for new kind of dialogues or behavior.

5.3 Repairing

An ordinary parser is a recognizer which means that it either *accepts* or *rejects* the input. We can not allow for rejecting any input since this would cause the dialogue component to fail. We instead have to repair the parse tree when faced with something ungrammatical. We currently use two principles which we will illustrate by showing how to process two turns (`fmw1_2_01` and `mps1_2_02`) taken from a German-German dialogue from our corpus. The assignment of the dialogue acts is provided by the semantic evaluation component. The translation sticks to the German words as close as possible and is not provided by VERBMOBIL. The trace of the dialogue component is given in figure 6.

...

`fmw1_2_01: der Termin den wir neulich abgesprochen haben am zehnten an dem Samstag` (MOTIVATE)
(*the date we recently agreed to, on 10th on that Saturday*)

`da kann ich doch nich'` (REJECT)
(*then I can not*)

`wir sollten einen anderen ausmachen` (INIT)
(*we should make another one*)

`mps1_2_02: wenn ich da so meinen Termin- Kalender anschaue,` (DELIBERATE)
(*if I look at my diary*)

`das sieht schlecht aus` (REJECT).
(*that looks bad*)

...

Figure 5: Two turns from an example dialogue

- **Statistical repair** In this example the statistical repair occurs when a rejection (the second REJECT) does not – as expected – follow a proposal (SUGGEST) (see figure

3). Instead, it comes after the introduction of the topic to be negotiated (INIT) and after a DELIBERATE. The latter dialogue act can occur at any point of the dialogue; it refers to utterances which do not contribute to the negotiation as such and which can be best seen as "thinking aloud". As first option, the plan recognizer tries to repair this state using statistical information, finding a dialogue act which is able to connect INIT and REJECT[4]. As can be seen in figure 6 the dialogue acts REQUEST_COMMENT, DELIBERATE, and SUGGEST can be inserted to achieve a consistent dialogue[5]. Ordered according to their scores, these candidates for insertion are tested for compatibility with either the previous or the current dialogue act. The notion of compatibility refers to dialogue acts which have closely related meanings or which can be easily realized in one utterance.

To find out which dialogue acts can be combined we examined the corpus for cases where the repair mechanism proposes an additional reading. Looking at the sample dialogues we then checked which of the proposed dialogue acts could actually occur together in one utterance, thereby gaining a list of *admissible* dialogue act combinations. In the VERBMOBIL corpus we found that dialogue act combinations like SUGGEST and REJECT can never be attributed to one utterance, while INIT can often also be interpreted as a SUGGEST therefore getting a typical follow-up reaction of either an acceptance or a rejection. The latter case can be found in our example: INIT gets an additional reading of SUGGEST.

- **Repair operators** In cases where no statistical solution is possible plan-based repair is used. When an unexpected dialogue act occurs a plan operator is activated which distinguishes various types of repair. Depending on the type of the incoming dialogue act specialized repair operators are used. The simplest case covers dialogue acts which can appear at any point of the dialogue, as e.g. deliberations (DELIBERATE) and clarification dialogues (CLARIFY_QUERY and CLARIFY_ANSWER). We handle these dialogue acts by means of repair in order to make the planning process more efficient: since these dialogue acts can occur at any point in the dialogue the plan recognizer in the worst case has to test for every new utterance whether it is one of the dialogue acts which indicate a deviation. To prevent this the occurrence of one of these dialogue acts is treated as an unforeseen event which triggers the *repair* operator.

6 Open Questions

We have presented the plan recognition component of the VERBMOBIL system: a robust and efficient implemented plan recognizer which copes with all of the over 200 annotated dialogues of the VERBMOBIL corpus. It is together with the rest of the dialogue component fully integrated in the first VERBMOBIL prototype. Future research

[4]Because DELIBERATE has only the function of "social noise" it can be omitted from the following considerations.

[5]The annotated scores are the product of the transition probabilities times 1000 between the previous dialogue act, the potential insertion and the current dialogue act which are provided by the statistic module.

```
 ...
 Planner: -- Processing REJECT
 Planner: -- Processing INIT
 Planner: -- Processing DELIBERATE
 Warning -- Repairing...
 Planner: -- Processing REJECT
 Warning -- Repairing...

Trying to find a dialogue act to bridge
DELIBERATE and REJECT ...

Possible insertions and their scores:
((SUGGEST 81326)
 (REQUEST_COMMENT 37576)
 (DELIBERATE 20572))

Testing SUGGEST for compatibility with
surrounding dialogue acts...

The previous dialogue act INIT
has an additional reading of SUGGEST:
INIT -> INIT SUGGEST !

 Planner: -- Processing INIT
 Planner: -- Processing SUGGEST
 ...
```

Figure 6: Trace of the two turns

concerns the following questions:

• **Incomplete input** The biggest problem with the VERBMOBIL setting is gaps in the input. Although the utterance is fully analyzed the semantic evaluation might not be able to assign a dialogue act to the utterance. Also, and more extremely, the output from the keyword spotter might not provide us with any information at all. Processing one missing input can be done using the predictions from the statistic component. When only one or two dialogue acts fail we can with fairly high reliability use the statistic component for suggestions. Our experiments show figures for up to 80% accuracy for predicting 3 dialogue acts. However, with larger gaps this technique will be insufficient.

• **Multiple input** Both the keyword spotter and the semantic evaluation will in the future produce multiple output annotated with probabilities.

- **Top down predictions** Our statistic based method for predicting the next dialogue act to come shows very promising results. How can the plan recognizer compete with and/or complement our statistical method.

References

[1] Jan Alexandersson, Elisabeth Maier, and Norbert Reithinger. A Robust and Efficient Three-Layered Dialog Component for a Speech-to-Speech Translation System. In *Proceedings of the 7th Conference of the European Chapter of the ACL (EACL-95)*, Dublin, Ireland, 1995. also available as Verbmobil Report Nr. 50, DFKI GmbH, Dezember 1994.

[2] Randall J. Calistri-Yeh. Utilizing User Models to Handle Ambiguity and Misconceptions in Robust Plan Recognition. *User Modelling and User-Adapted Interaction, Special Issue on Plan Recognition*, 1991.

[3] Susanne Jekat, Alexandra Klein, Elisabeth Maier, Ilona Maleck, Marion Mast, and J. Joachim Quantz. Dialogue acts in verbmobil. Verbmobil Report 65, Universität Hamburg, DFKI Saarbrücken, Universität Erlangen, TU Berlin, 1995.

[4] Marion Mast. Schlüsselwörter zur Detektion von Diskontinuitäten und Sprechhandlungen. Technical Report Verbmobil Memo Nr. 57, Friedrich-Alexander-Universität, Erlangen-Nürnberg, January 1995.

[5] Marc Vilain. Getting Serious about Parsing Plans: a Grammatical Analysis of Plan Recognition. In *Proceedings of AAAI-90*, pages 190–197, 1990.

[6] Wolfgang Wahlster. Verbmobil – Translation of face-to-face dialogs. In *Proceedings of the Fourth Machine Translation Summit*, Kobe, Japan, July 1993.