From J. Haas et al., *TSD'98, Int. Workshop on Text, Speech and Dialogue*, Brno (Czech Republic), 1998, (pp. 357–362).

A Concept for a Prosodically and Statistically Driven Chunky Semantic Parser

Jürgen Haas, Manuela Boros, Elmar Nöth, Volker Warnke, Heinrich Niemann

University of Erlangen-Nürnberg
Chair for Pattern Recognition – Martensstraße 3 – 91058 Erlangen – Germany
(haas,boros,noeth,warnke,niemann)@informatik.uni-erlangen.de

Abstract. In spoken dialog systems typically just a small set of predefined information has to be provided to the system in order to accomplish its task. We present here a concept for a partial (chunky) semantic parser, whose task is to detect the parts in a user utterance that contain needed information and to analyze these parts linguistically. In order to support the parser, we introduce statistical and prosodic methods to predict semantic concepts in an utterance und their location.

1 Introduction

Following the most common architecture of spoken dialog systems as shown in Figure 1, the main task of linguistic processing is to yield a semantic representation of what the user said. These representations are interpreted by the dialog module according to the dialog context and the system answer will be generated. The system utterance depends on whether the system still needs certain information or if all necessary information has been given to accomplish its task. The dialog module keeps track of the relevant semantic concepts uttered by the user and therefore knows when all required information has been provided.

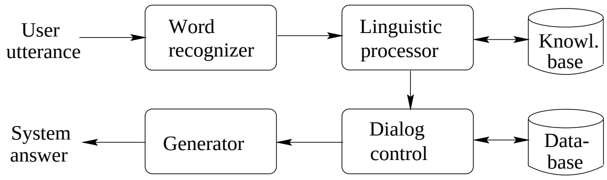

Fig. 1. Typical dialog system architecture.

Linguistic processing has to cope with unrestricted (spoken) text, that most often is ungrammatical. However, to yield a successful dialog only the semantically relevant parts of user utterances are important, e.g. only those concepts, that must be interpreted. Therefore, linguistic analysis could be restricted to those parts. In our case of a train timetable information, values for the semantic slots *time, goal, source*, and *date* have to be filled in order to perform the database query. We therefore suggest that linguistic analysis should be restricted to these concepts, resulting in several grammar fragments. As a result, the system's linguistic database (grammar) will be much smaller and more modular, allowing for easy maintenance and reuse in other domains. Since parts of the utterances can be neglected during parsing, the processing time is expected to be much shorter than for a full covering grammar. In section 2 we will describe the used grammar formalism and first experiments.

For this kind of linguistic analysis, still one question remains: Which concepts do we have to look for in an utterance? Most often the expected concepts can be predicted due to the last system utterance. E.g. if the system asked *When do you want to leave?* the answer is expected to contain a time expression. However, the user not necessarily answers with specifying a time. He also may correct misunderstandings or provide additional information. In such cases the parser needs to know which concepts it has to look for. In order to provide this information we use statistical methods to compute probabilities of each concept to occur in the utterance. This prediction technique is described in section 3.

However, even if the parser knows, which grammar fragment to apply, when dealing with word hypotheses graphs, it may find several competing concept tokens. In order to increase the probability of detecting the correct user utterance, word hypotheses graphs may contain more than one path, each of them representing one potential user utterance. The parser, e.g. when looking for time expressions, has to search the complete graph to get all alternatives, and must choose one upon them. The choice among alternatives is driven by acoustic scores. In order to detect the correct (partial) path most quickly, analysis should start at the correct alternative already. We use prosodic information providing plausibility scores, which are merged with acoustic scores, thus improving scoring of hypotheses. Computation of prosodic scores is described in section 4.

The resulting system, integrating partial parsing, statistic concept prediction and prosody, is expected to work very efficiently on either word strings or lattices and can replace the full linguistic analysis as used in our system so far.

2 The Chunky Parser

The parser's task is to locate and analyze those parts of a user utterance that have to be interpreted by the dialog module. In this sense, it behaves like a chunk parser, whose aim is to find all chunks in a given sentence without necessarily attaching them to a complete analysis. The main differences between a chunk parser and our chunky parser are, that we locate the relevant parts by ignoring the rest and that the *chunks* we look for are motivated semantically only.

The parser of our system is an unification based island driven chart parser, that can handle either word strings or word hypotheses graphs. In case of word graphs, the island parsing strategy is applied by starting the analysis at several points in the graph, trying to expand these *islands* successively to the left and the right. In either way, the parser may start at an arbitrary point. For the detection and analysis of semantic chunks, possible surface forms are described by a grammar fragment per concept. The linguistic knowledge base therefore does not contain one full grammar, but several grammar modules, that may be grouped individually together if the application changes. The grammars for the chunky parsers are developed in terms of a context-free phrase structure grammar that comprises lexica and bundles of context-free grammar rules. Each linguistic sign is described through a complex feature structure covering morphologic, syntactic, and semantic information. As the aim is to yield a semantic representation, most emphasis lies on the semantic representation in the feature structure. As an example the semantic representation of the sentence *I want to go to Munich.* is given on the left side in figure 2. The chunky parser only searches for and analyses semantically relevant parts, e.g. the part *to Munich* as realization of the concept *goal*. The resulting representation of *I want to go to Munich.* is shown on the right side in figure 2.

$$\begin{bmatrix} type : go & \\ thegoal : & \begin{bmatrix} type : location & \\ thecity : & \begin{bmatrix} type : city \\ value : munich \end{bmatrix} \end{bmatrix} \\ thetheme : & \begin{bmatrix} type : individual \\ thediscourserole : \\ \qquad \begin{bmatrix} val : speaker \end{bmatrix} \end{bmatrix} \end{bmatrix} \qquad \begin{bmatrix} type : location & \\ thecity : & \begin{bmatrix} type : city \\ value : munich \end{bmatrix} \end{bmatrix}$$

Fig. 2. Full (left) and partial semantic representation (right).

2.1 Experiments

Our first grammar fragment covers the semantic concepts `date` and `time`. Typical time expressions in our system are hours (*at nine o'clock*), dates (*on the first of July*), weekdays (*on Monday*), or relative dates (*tomorrow*). Grammar fragments are developed by collecting sample data and choosing some representative examples from it. First experiments were run on a corpus of utterances collected with our system over the public telephone network. A set of 119 utterances was chosen from the corpus covering only time expressions, containing 389 words which yield 2618 word hypotheses in the resulting graphs. Semantic accuracy was measured using semantic concept accuracy (CA) following [1]. For comparison we also parsed these utterances with our full grammar. Experiments were run on word hypotheses graphs and on transcriptions. of the utterances. Table 1 shows the resulting figures; results on the transcriptions are marked with *script*, results on word graphs with *graph*.

	Full Grammar	Fragment		Full Grammar	Fragment
time script	9.2 s	4.6 s	time graph	67.5 s	18.0 s
CA script	98 %	89 %	CA graph	59 %	68 %

Table 1. First experiments with a Grammar Fragment.

The semantic coverage of the full grammar is very good, whereas the time grammar does not yet achieve the same coverage. CA on the word graphs decreases for the full grammar more than for the fragment. Also processing time is much higher for the full grammar especially for word graphs. The reason lies in the fact that the chunky parser using a grammar fragment just searches the hypotheses graph for paths belonging to the expected semantic concept. The full grammar has to search the full graph to find a complete path, which also increases the possibility of following the wrong path resulting in a loss in CA.

3 The Semantic Concept Predictor

We examine a statistical approach using n-gram language models as semantic concept predictor. The model has to decide about the occurrence of semantic concepts in word chains. We prove its usability on the above mentioned corpus containing the 119 utterances for the grammar development. The predictor decides whether there is a time expression or not. The method then can be extended to other semantic concepts like *goal, source*, and *date*.

The language model computes estimations for the occurrence of a word w_i under the assumption of its predecessor words $w_{i-1}, \ldots, w_{i-n+1}$ as the linear interpolation of the n-gram and all smaller n-grams to smooth the probabilities. The interpolation weights ρ_i are used as weighting factors and are estimated automatically with a cross validation technique [2]. The probability $p(\boldsymbol{w})$ for a word chain $w_1 \ldots w_m$ is given by the equation:

$$p(\boldsymbol{w}) = \prod_{k=1}^{m} \left(\rho_0 \cdot \frac{1}{L} + \rho_1 \cdot p(w_k) + \sum_{i=2}^{k} \rho_i \cdot p(w_k \mid w_{k-i+1} \ldots w_{k-1}) \right) \qquad (1)$$

Using n-gram language models as a semantic concept predictor we have to claim for a word chain $\boldsymbol{w}$ whether the concept we are looking for is expressed in $\boldsymbol{w}$ or not. Therefore we build two language models, one is trained with word chains expressing the semantic concept and the other with utterances not giving it. During analysis we compute the two scores for the incoming word chain – when using word graphs we choose the best word chain in the graph – and decide for the higher probability. A more detailed analysis is possible when we split the training data into three different sets. The first comprises all word chains where the concept does not appear, the second one all utterances where only the concept is expressed and no other semantically relevant information is present and the third set all word chains where the interesting semantic concept appears along with additional information. The decision rule again decides for the highest probability of the three scores.

3.1 Experiments

For training and test purposes we use 10114 sentences collected with the above mentioned system. As we concentrate on the detection of time expressions we mark, based on the transliteration, each sentence whether there is the semantic concept *time* or not (NO), and if it is there whether there appears only this (ONLY) or even more task relevant information (PLUS). The available data is split 2/3 to 1/3 for training and test. The number of sentences for each class is presented in Table 2. Since a word sequence from the test set might have been used by a different speaker from the training set, the column 'test $\neq$ train' gives the number of sentences from the column 'test' that were not observed during training. We train 5-grams for the three classes and use linear interpolation of the scores. The 'Semantic Concept Predictor' results we obtain are reported in Table 2. We see that our approach to the prediction task performs quite well and could therefore be used as a predictor for the semantic concept analysis. For the two class problem we obtain a recall of 97.8% and a precision of 56.9% for the spoken word chain and 72.4% recall and 49.0% precision for the recognized.

			test $\neq$ train		spoken word chain				recognized word chain			
	train	test	spoken	recog.	ONLY	PLUS	NO	RR	ONLY	PLUS	NO	RR
ONLY	92	45	16	17	**91.1**	8.9	0.0	**85.3**	**71.1**	11.1	17.8	**81.1**
PLUS	1070	535	313	375	8.8	**88.8**	2.4		6.0	**65.6**	28.4	
NO	5582	2790	874	1227	7.2	8.2	**84.6**		6.9	8.8	**84.3**	

Table 2. Number of word chains, detection rates with the spoken and recognized chain

4 Prosodic Scoring

The use of prosodic information for spoken dialog systems becomes more and more important. In the VERBMOBIL-project [3] prosodic information was successfully used in a speech understanding systems for the first time. Using a neuronal network (NN) with different sets of prosodic features like pitch-contour, energy-contour and duration measures, phrase boundaries, sentence mood and phrase accent are determined [5].

We use this prosodic information to determine the focused regions in a phrase, which are those parts that hold the most important content words. To get information for the focused regions, we use a NN trained on a part of the VERBMOBIL-database with 276 nodes in the input-layer (one node per prosodic feature), and 2 nodes in the output-layer (accentuated A vs. not ¬A) and one hidden-layer. Using the scores $Score(A \mid w)$ and $Score(\neg A \mid w)$ we estimate the probability $P(\mathsf{A} \mid w) = \frac{Score(\mathsf{A}|w)}{Score(A|w)+Score(\neg A|w)}$ and decide for a focused region with a threshold. This knowledge can be used with our parser in two ways. The first is to rank the regions by their prosodic scores and offer this list to the parser. The second is to get a list of possible expressions from the parser and disambiguate them using the prosodic scores. Both ways can efficiently be used to find the best expression the parser is searching for in the context the concept predictor has estimated. Working on word hypotheses graphs the first way is better as the parser only searches in the best scored paths and search effort is smaller.

4.1 Experiments

In this section we present results for determining stressed words for different dialog acts (see [4]) in the VERBMOBIL-database using the above described NN. In VERBMOBIL there are 42 illucotionary dialog acts grouped into 18 classes. For these we estimated the most frequent stressed words of a subset of the VERBMOBIL-database. For this approach we only ranked those words, whose stress probability exceeds a threshold of 0.8 and were seen stressed in more than 80% of their occurrences. In Table 3 one can see the ten most often seen automatically estimated stressed words for all dialog act classes together along with the five most often seen detected stressed words for the most frequent dialog act classes SUGGEST and ACCEPT. In both tables the words are ranked by their frequency of occurrence. The results show, that we are able to detect the content words of an utterance if we determine the stressed words. This fact is very important for the use of this method to estimate the focused regions by only using acoustic features to decide for semantically important information.

All dialog acts		
Rk.	*% str.*	*word (translation)*
1	88.57	Freitag (Friday)
2	82.69	Wiederhören (bye)
3	84.31	Donnerstag (Thursday)
4	90.91	Samstag (Saturday)
5	95.35	neunzehnten (19th)
6	81.82	August (August)
7	96.15	vierundzwanzig. (24th)
8	87.50	achten (8th)
9	86.96	wunderbar (marvellous)
10	100.00	sechsundzwanzig. (26th)

ACCEPT		
1	100.00	einverstanden (ok)
2	100.00	Ordnung (alright)
3	100.00	wunderbar (marvellous)
4	85.71	Freitag (Friday)
5	85.71	frei (free)

SUGGEST		
1	82.22	Montag (Monday)
2	87.80	Freitag (Friday)
3	83.33	Donnerstag (Thursday)
4	82.76	Mittwoch (Wednesday)
5	93.10	Samstag (Saturday)

Table 3. Automatically determined stressed words

5 Conclusion and Further Work

In our paper we presented a concept for partial semantic parsing in a spoken dialog system. The analysis is restricted to those parts of utterances that contain semantic concepts necessary for understanding. The parser acts like a chunk parser as it looks for subparts in the input string. These subparts correspond to semantic concepts and are not necessarily adjacent to each other, i.e. parts of the sentence may stay unreviewed. Syntactic analysis is guided by grammar fragments that are activated by a statistical predictor that can tell which concepts occur in an utterance. This information is supported by prosodic scores that assign higher probability to words that belong to predicted concepts. Each of the subparts has been tested in preliminary experiments that show encouraging results. Next we plan to extend the number of grammar fragments and to integrate the system in order to verify our assumptions on accuracy and efficiency.

References

[1] M. Boros, W. Eckert, F. Gallwitz, G. Hanrieder, G. Görz, H. Niemann. 1996. Towards understanding spontaneous speech: Word accuracy vs. concept accuracy. In *Proceedings of ICSLP'96*, pp. 1005-1008, Philadelphia, 1996.

[2] H. Ney, S. Martin, F. Wessel. 1997. Statistical Language Modeling Using Leaving-One-Out. In *Corpus-based Methods in Language and Speech Precessing*, pp. 210-234. Kluwer Academic Publishers. Boston.

[3] T. Bub and J. Schwinn. 1996. Verbmobil: The Evolution of a Complex Large Speech-to-Speech Translation System. In *Int. Conf. on Spoken Language Processing*, volume 4, pages 1026–1029, Philadelphia, 1996.

[4] S. Jekat, A. Klein, E. Maier, I. Maleck, M. Mast, and J. Quantz. 1995. Dialogue Acts in Verbmobil. Verbmobil Report 65, April 1995.

[5] A. Kießling. 1997. *Extraktion und Klassifikation prosodischer Merkmale in der automatischen Sprachverarbeitung.* Berichte aus der Informatik. Shaker Verlag, Aachen.

This article was processed using the LaTeX macro package with LLNCS style