From J. Haas et al., *SQEL'98 Workshop on Multi-Lingual Information Retrieval Dialogs*, Stara Lesna (Slovakia), 1998, (pp. 422–430).

A Probabilistic Approach for the Semantic Analysis

J. Haas, J. Hornegger, E. Nöth, H. Niemann

Universität Erlangen-Nürnberg,
Lehrstuhl für Mustererkennung (Informatik 5)
Martensstraße 3, D-91058 Erlangen
Germany
email: (haas,hornegger,noeth,niemann)@informatik.uni-erlangen.de

1 Introduction

For the task of speech understanding and automatic dialog systems it is not that important to recognize the user's utterance with 100% word accuracy but to get the user's intention in order to generate an appropriate system reaction. In our application of information retrieval dialogs possible reactions are e.g. to start a clarification subdialog or to give the desired information. Nowadays the syntactic-semantic analysis is mainly realized using rule based systems. These parsers use a bunch of lexical and grammatical knowledge. Hence, the system adaptation to another domain or task requires the encoding of new lexical and grammatical knowledge nearly from scratch. This is a time consuming and tricky work which has to be done by a linguistic expert. Having a probabilistic model for the semantic analysis which is able to learn the new task from a suitable training set a faster and easier adaptation is possible.

The paper is organized as follows. In section 2 we present the probabilistic model for the semantic analysis and the mathematical details. The semantic annotation scheme allowing a fast and easy creation of training material is introduced in section 3. There we also describe the data we use and the baseline experiment with its results. The supporting methods for our model are presented in section 4. The reported results prove their feasibility individually and in combination. In section 5 we give a short summary and an outlook on planned activities.

2 Probabilistic Model for Semantic Analysis

Our probabilistic model is based on the assumption that for each word w_j of a word chain $\underline{w}$ we have an assignment of one semantic attribute C_i to w_j. These

This work was partly funded by the European Community in the framework of the SQEL–Project (Spoken Queries in European Languages), Copernicus Project No. 1634. The responsibility for the contents lies with the authors.

semantic attributes describe parameters which influence the system's reaction. The set $\mathcal{C} = \{C_1, C_2, \ldots C_N\}$ of possible attributes is therefore strongly task dependent. Obviously not all parts of the word chain are important for its interpretation. To keep our assumption we have to introduce an additional semantic attribute C_0, the NIL attribute. All words carrying no meaning in the domain are attached to C_0. An example illustrating this assumption is given in figure 1.

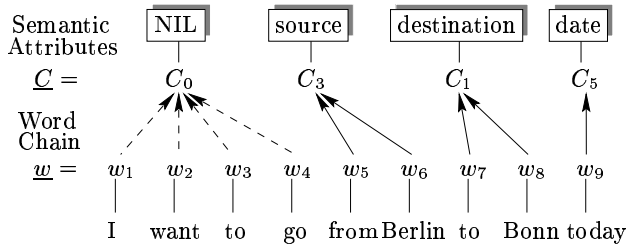

Figure 1. Assignment of semantic attributes C_k to words w_j

Each semantic attribute C_i is now identified with a random process that generates words w_j. Each such process defines a density function $P(w_j|C_i)$ over the lexicon of words. The assignment of a semantic attribute C_i to a word w_j is formally described with the assignment function ζ. This function takes as argument the word w_j of the word chain $\underline{w}$ and gives as result the index of the corresponding attribute $\zeta(w_j) = l_j$. As we want to analyze word chains $\underline{w}$ we have to extend the assignment function:

$$\underline{\zeta}(\underline{w}) = (\zeta(w_1), \zeta(w_2), \ldots, \zeta(w_n))^T = (l_1, l_2, \ldots, l_n)^T \tag{1}$$

The semantic of a word chain $\underline{w}$ in the current task is now described through the set of semantic attributes $\underline{C}$ that are assigned to the words of $\underline{w}$. All words corresponding to one attribute from $\underline{C}$ constitute one semantic segment in the word chain. In comparison to [3] who also defines an assignment from the english to a formal language for the interpretation of word chains, our approach does not make use of the length of semantic segments or the number of assigned attributes for the computation of probabilities. The statistic parameters characterizing this mapping are estimated using the EM algorithm.

The semantic analysis of word chains is now reduced to the problem of finding the best set of semantic attributes that are assigned to the words of $\underline{w}$. To provide a formalism which enables us to handle the speech understanding task statistically we introduce a model based on two tied probabilistic processes. The first one has to model the appearance of semantic attributes and introduces a probability distribution on the set of possible assignments $\underline{\zeta}(\underline{w})$. The second one is the semantic attribute generating words with the density function $P(w_j|C_i)$. With these two processes the probability $P(\underline{w}, \underline{\zeta}|\underline{C})$ for observing $\underline{w}$ and the assignment $\underline{\zeta}$ assuming the semantic described through $\underline{C}$ is computed. The assignment vector can not be observed as we have only the set of semantic attributes

for a word chain but not the explicit alignment. Thus, the probability $P(\underline{w}|\underline{C})$ derives from the marginalization of $P(\underline{w},\underline{\zeta}|\underline{C})$ over all possible assignments, i.e

$$P(\underline{w} \mid \underline{C}) = \sum_{\underline{\zeta}} P(\underline{w},\underline{\zeta}|\underline{C}) = \sum_{\underline{\zeta}} P(\underline{\zeta}) P(\underline{w} \mid \underline{C},\underline{\zeta}) \ . \tag{2}$$

We now claim that assignments depend only on preceding ones and are of statistical order g. Then $P(\underline{\zeta})$ is factorized in conditional probabilities $p(l_k \mid l_{k-g} \ldots l_{k-1})$. Renaming these probabilities with $p(l_k \mid l_{k-g} \ldots l_{k-1}) = a_{l_{k-g} \ldots l_k}$ we get the following:

$$\begin{aligned} P(\underline{\zeta}) &= p(l_1, l_2, \ldots, l_n) \\ &= p(l_1) \cdot p(l_2 \mid l_1) \cdot p(l_3 \mid l_1 l_2) \cdot \ldots \cdot p(l_n \mid l_1 l_2 \ldots l_{n-1}) \\ &= p(l_1) \cdot p(l_2 \mid l_1) \cdot \ldots \cdot p(l_g \mid l_1 l_2 \ldots l_{g-1}) \cdot \prod_{k=g+1}^{n} p(l_k \mid l_{k-g} \ldots l_{k-1}) \\ &= a_{l_1} \cdot a_{l_1 l_2} \cdot \ldots \cdot a_{l_1 l_2 \ldots l_g} \prod_{k=g+1}^{n} a_{l_{k-g} \ldots l_k} \end{aligned} \tag{3}$$

The probability $P(\underline{w} \mid \underline{C},\underline{\zeta})$ is reduced to probabilities for the observation of single words w_j depending only on the actual assignment to the semantic attribute C_{l_j}. With these assumptions the probabilistic semantic analysis problem is reformulated:

$$\begin{aligned} P(\underline{w} \mid \underline{C}) = & \sum_{\underline{\zeta}} p(\underline{\zeta}) \prod_{j=1}^{n} P(w_j \mid C_{\zeta(w_j)}) \\ = & \sum_{l_1, l_2, \ldots, l_n} a_{l_1} \cdot a_{l_1 l_2} \cdot \ldots \cdot a_{l_1 l_2 \ldots l_g} \prod_{k=g+1}^{n} a_{l_{k-g} \ldots l_k} \cdot \prod_{j=1}^{n} P(w_j \mid C_{l_j}) \end{aligned} \tag{4}$$

The parameters to be estimated for the stochastic model are the conditional probabilities for the elements of the assignment vector $a_{l_{k-g}, l_{k-g+1}, \ldots, l_k}$ and the probabilities $P(w_j|C_{\zeta(w_j)})$ for observing word w_j under the assumption of assigning it to the attribute $C_{\zeta(w_j)}$. As the assignment $\underline{\zeta}$ is hidden in our training data, we use the Expectation–Maximization algorithm (cf. [1]). As result we obtain iterative estimation formulas for the necessary parameters. These estimation formulas can be found in [5, 6]. Setting the statistical dependency to order $g = 1$ the well known Baum–Welch re-estimation formulas for HMM are derived. If we increase the statistical order g a generalized version of HMM is formalized. Probabilities are computed with a generalized forward-backward algorithm. The most likely state sequence, which gives the semantic segmentation for a word chain $\underline{w}$ and along with that the set of semantic attributes $\underline{C}$, results from a generalized Viterbi algorithm.

3 Annotation Scheme and Baseline Experiment

The data we use for our experiments is collected with the automatic dialog system EVAR. This system answers questions on train timetables for German Intercity connections and is hooked to a public telephone line (Tel. ++49/9131/16287). The EVAR system and the data are described in [2]. In this section we first describe the annotation scheme we use which is a quick and easy method to create the necessary training material. In the second part we report about the baseline experiment and the obtained recognition results.

All dialogs performed by EVAR are recorded and information like the recognized word chain or the text of the synthesized answer is stored. The speech signals are then transcribed off-line, i.e. it is annotated what really was spoken. The semantic we annotate is derived from this spoken word chain. We defined a set of about 30 different semantic attributes to be distinguished in the users' utterances. We consider only those parameters as semantic attributes which influence the system's reaction. For example time and date expressions are important for the system to give the correct information. Also positive or negative feedback are regarded as semantic attributes because of their influence on the ongoing dialog. In contrast the caller's name or the reason why someone wants to travel by train is obviously unimportant and therefore neglected in the semantic annotation. The resulting semantic is independent of the dialog context. The same word sequence $\underline{w}$ has always the same semantic annotation regardless of the yet collected information or the actual system dialog step. For example if the system asks `'Where do you want to leave?'` often an answer with a city name e.g. `'Hamburg'` is observed. Then only `'city:hamburg'` is annotated as being the semantic content. Taking the dialog context into account it is obvious that the named city is the departure city (`'sourcecity:hamburg'`). The semantic annotation for a spoken word chain $\underline{w}$ is the semantic described as the set of semantic attributes $\underline{C}$. As we do not need the explicit alignment of semantic attributes to words this annotation can be done without linguistic knowledge. This approach of describing the semantic is similar to the one described in [7] with semantic concepts.

For our experiments the 30 different semantic attributes we annotate are clustered together in the following twelve classes:

1.	**SOURCECITY**	departure city
2.	**GOALCITY**	arrival city
3.	**CITY**	cities not in 1 or 2
4.	**MARKER**	parts with influence on dialog e.g. yes
5.	**RELDAY**	dates in relative manner e.g. tomorrow
6.	**SPECIAL**	legal holidays e.g. Christmas
7.	**DATE**	dates not in 5 or 6
8.	**TRAINTYP**	type of train e.g. IC or ICE
9.	**POFDAY**	time as part of day e.g. in the morning
10.	**RELTIME**	time in relative manner e.g. later
11.	**TIME**	time not in 9 or 10
12.	**NIL**	parts semantically irrelevant

As training set we have 10207 sentences along with their semantic annotation. Due to the dialog strategy some of the sentences are very frequent. For example, the requested confirmation of parameters for the database query are very often answered with a `'yes'` or `'no'`. In the training data there are 3977 different sentences. As test set we choose another 4952 sentences collected with EVAR where 2539 are different. Some sentences in the test set are also included in the training set but 2421 sentences are unseen. In training and test set there are 1682 different words which serve as observables for the probabilistic model.

Each attribute to be recognized is represented by one state in our probabilistic model. All states are connected with each other. If we set the statistical dependency to $g = 1$ we have an ergodic HMM. The output probabilities are discrete distributions over the words in the lexicon. As the EM algorithm has only local optimality and a slow convergence we have to choose the initialization of the probabilities for our introduced method carefully. The transition probabilities representing the conditional probabilities for the assignment function are uniformly initialized for the baseline experiment. The output probabilities are simply relative frequencies of words' occurrences. We take all sentences from the training set where the modeled semantic attribute is observed and count the words therein. We start to counting at 1 to avoid a probability of value 0.

Using this modeling technique and the described initialization methods we obtain the recognition results shown in Table 1. The percentages give the recognition accuracy, i.e. we count all sentences as error where either an attribute is deleted or inserted. Therefore the number of deletions plus the number of insertions is higher than the number of wrong sentences. If we have both, a deletion and an insertion in one word chain, we only have one erroneous sentence. Within this evaluation scheme the question arises if an insertion error should be judged as expensive as a deletion because in the next step after recognizing the semantic attributes we have to extract the corresponding values (e.g. the real spoken city name for the database query). In this step an insertion error can be discovered and repaired but a deletion error not. In the moment we are counting deletions and insertions to scale our model's performance.

	HMM	G2HMM
Accurate sentences	2843	2857
Wrong sentences	2109	2095
Accurate Detections	57 %	58 %
	Insertions	
Sentences	2068	2068
	Deletions	
Sentences	663	671

Table 1. Accuracy and number of errors of the baseline experiment with statistical dependency of order $g = 1$ (HMM) and order $g = 2$ (G2HMM) with twelve semantic attributes

4 Methods and Results

The baseline experiment proves the capability of our approach to perform the task of semantic analysis. In this section we introduce and examine some methods which support our statistical model in semantic analysis. Employing a method on its own gives only small improvement in recognition rate. The impressive supporting power of these methods turns out when they are applied in combination.

4.1 Categorial System

The first method we use is well-known in language modeling and estimating word occurrence probabilities from small data sets. We use a system of categories which partitions the lexicon, i.e. each word belongs to one and only one category. These categories are built considering syntactic and semantic constraints. In our experiments we use two different categorial systems, the first one with 220 categories which serve as observables for the statistical model. For the second system we only use 13 different main categories taken from the first categorial system like `'CITYNAME'` or `'WEEKDAY'`. The words that do not belong to one of these categories build their own category. This output alphabet consists of 1089 observables. We still initialize the transition probabilities uniformly and the output probabilities as relative frequencies, but we reduced the size of the lexicon. For the experiments with category systems the results are shown in Table 2.

	220 Observ.		1089 Observ.	
	HMM	G2HMM	HMM	G2HMM
Acc. sent.	3111	3122	3072	3093
Wrong sent.	1841	1830	1880	1859
Acc. Detec.	63 %	63 %	62 %	62 %
	Insertions		**Insertions**	
Sentences	1814	1806	1845	1836
	Deletions		**Deletions**	
Sentences	338	350	350	323

Table 2. Accuracy and number of errors of the experiments using categorial systems

4.2 Initial Output Probabilities

In the next experiment we change the initialization method for the output probabilities. As described in section 3 we use relative frequencies of words by counting the words that are observed besides a special attribute and we start to count at 1. In this approach we start to count at 0 because if a word w_j is never seen in combination with the attribute C_i we want to avoid explicitly the assignment of this attribute in the analysis phase. Giving the word w_j the output probability

$P(w_j \mid C_i) = 0$ realizes this. The recognition results of this experiment using again the complete word lexicon of 1682 words without categories are shown in Table 3.

	HMM	G2HMM
Accurate sentences	2918	2913
Wrong sentences	2034	2039
Accurate Detections	59 %	59 %
	Insertions	
Sentences	1973	1956
	Deletions	
Sentences	778	825

Table 3. Accuracy and number of errors of the experiments starting the count for the relative frequencies for the initial output probabilities at 0.

4.3 Initial Transition Probabilities

When we take a brief look at the results obtained in the experiments described in sections 4.1 and 4.2 we see that a lot of insertion errors occur. This is partially due to the fact that our transition probabilities are uniformly initialized. Then the decision for the best assignment is only guided by the output probabilities. Additionally, the EVAR database proves that attributes have a minimum length in number of words that is needed to express the attribute. For example, it is impossible to express a departure city by only one word which then has to be the name of the city. There must be at least one additional word (e.g. `'from'`) indicating the special semantic meaning. Therefore, attributes have characteristic length distributions counted in number of words. We could introduce some simple length modeling in our statistical model by initializing transition probabilities not uniformly but using these distributions. The results of this initialization scheme are reported in Table 4. Obviously the improvement is due to the fact that the number of insertions is decreasing.

4.4 Combinations

The introduced methods are now combined together to get the optimal recognition results in the borders of the yet examined methods. The best accuracy rate we obtain is reported in Table 5. For this experiment we:

- start counting for relative frequencies at 0,
- use the categorial system with 220 classes,
- use initial transition probabilities for length modeling.

	HMM	G2HMM
Accurate sentences	3164	3124
Wrong sentences	1788	1828
Accurate Detections	64 %	63 %
	Insertions	
Sentences	1299	1751
	Deletions	
Sentences	1347	910

Table 4. Accuracy and number of errors of the experiments with initial transition probabilities used for a crude length modeling of attributes.

	HMM	G2HMM
Accurate sentences	3652	3455
Wrong sentences	1300	1497
Accurate Detections	74 %	70 %
	Insertions	
Sentences	812	1374
	Deletions	
Sentences	890	835

Table 5. Accuracy and number of error of the experiments with the best recognition results.

5 Conclusion and Future Work

We present a probabilistic approach for the semantic analysis of word chains and several methods which support the statistical model and improve the performance of the analysis. The problem of finding the semantic of a word chain is formulated in a probabilistic manner. Our statistic model is based on an unknown assignment function for which the statistical parameters are estimated with the EM algorithm depending on the desired statistical dependency. For a dependency of order $g = 1$ the well-know HMM formalism results, increasing g gives generalized versions of HMMs.

We introduce our annotation scheme for the semantic content of a word chain with so called semantic attributes. In the baseline experiment we prove the feasibility of our statistical model. We achieve a semantic attribute accuracy rate of 57% for a statistical dependency of order 1 and 58% for order 2. The following section presents several methods for improving the performance of our model and show their usability. We test categorial systems on the lexicon and several methods for initializing the output and transition probabilities. The methods give only small improvements when applied uniquely but their combination shows impressive results. We achieve the best recognition results of

74% for order $g = 1$ and 70% for order $g = 2$ with the categorial system with 220 classes, initial transition probabilities modeling the length of attributes and starting the count of relative word frequencies at 0.

In the near future we are concentrating on the initial output probabilities. As our statistical model is very sensitive about this initialization we are looking for better methods to be used e.g. 'intelligent' smoothing techniques on the relative frequencies. We are also considering an information theoretical measure like the salience introduced in [4] for a smoothing operation on the output probabilities. Besides that we want to use the semantic annotation as additional knowledge during the training. In the moment we use it only for the initialization of output probabilities.

References

1. A. Dempster, N. Laird, and D. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society, Series B (Methodological)*, 39(1):1–38, 1977.
2. W. Eckert, E. Nöth, H. Niemann, and E. Schukat-Talamazzini. Real Users Behave Weird — Experiences made collecting large Human–Machine–Dialog Corpora. In *Proc. of the ESCA Tutorial and Research Workshop on Spoken Dialogue Systems*, pages 193–196, Vigsø, Denmark, June 1995.
3. M. Epstein, K. Papineni, S. Roukos, T. Ward, and S. Della Pietra. Statistical natural language understanding using hidden clumpings. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 176–179, Atlanta, 1996.
4. A. Gorin. On automated language acquisition. *Journal of the Acoustic Society of America*, 97(6):3441–3461, 1995.
5. J. Haas, J. Hornegger, R. Huber, and H. Niemann. Probabilistic semantic analysis of speech. In E. Paulus and F. M. Wahl, editors, *Mustererkennung 1997, DAGM–Symposium*, pages 270–277, September 1997.
6. J. Hornegger. *Statistische Modellierung, Klassifikation und Lokalisation von Objekten.* Phd–Thesis, Technical Faculty, University Erlangen-Nürnberg, 1996.
7. R. Pieraccini and E. Levin. A learning approach to natural language understanding. In *NATO-ASI, New Advances & Trends in Speech Recognition and Coding*, volume 1, pages 261–279, Bubion (Granada), Spain, 1993.