Semantigrams — Polygrams Detecting Meaning

Jürgen Haas and Elmar Nöth and Heinrich Niemann

Universität Erlangen-Nürnberg,
Lehrstuhl für Mustererkennung (Informatik 5)
Martensstraße 3, D-91058 Erlangen, F.R. of Germany
E-mail: haas@informatik.uni-erlangen.de

Abstract. In this paper we present a statistical approach to shallow linguistic analysis of word chains modeled with probabilistic methods. We reduce the linguistic analysis to the problem of extracting pairs of semantic attributes and corresponding values. This problem can be split up in two parts: (1) detection of semantic attributes in the sentence and (2) extraction of corresponding values to the attributes. Here, we concentrate on the first problem and we use the well known polygram language models to determine the semantic attributes included in a word chain. We train different models for different sets of semantic attributes and apply a maximum–likelihood decision rule on these probabilistic measures. The results will show, that the idea of modeling the appearance of different meanings in a word chain with attribute–dependent language models is successful.

1 Introduction

The efficiency of dialog systems depends heavily on the quality of the linguistic analysis. A robust semantic decoding of incoming speech should be able to cope with errors made in the word recognition module. It is crucial for a successful dialog that the intention of the user is correctly recognized so that an adequate system reaction is produced. As the history of pattern recognition tells us, statistical models often perform better than knowledge-based approaches or heuristic solutions.

Up to now in existing automatic dialog systems the usage of statistics within the complete systems is established in the word recognition module using an HMM-based approach for acoustic modeling and n-gram language models for describing syntactical constraints. Only a few systems try to go beyond recognition with probabilistic methods e.g. [2, 4].

In this paper we present an approach to shallow linguistic analysis by using polygram language models. In the first step, which is reported here, we try to extract the semantic of a word chain by detecting semantic attributes. We assume that the semantic content of a word chain $\boldsymbol{w}$ can be represented by pairs of semantic attributes and their values. We evaluate our probabilistic approach in the domain of train-time-table information retrieval dialogs. Therefore, the interesting semantic attributes are e.g. *sourcecity*, *goalcity*, *time of arrival*. For example, if we want to describe the semantic content of the word chain

'Ich will von Berlin nach Hamburg'
'I want to go from Berlin to Hamburg'

a correct description in our domain should look like this:

[sourcecity:Berlin] [goalcity:Hamburg] .

Our goal is to get the attributes *sourcecity* and *goalcity* and nothing else as a result when classifying the word chain mentioned above with stochastic models.

2 Polygram Language Models

Stochastic language models are used for modeling language constraints and are applied in an efficient way in speech recognition [3]. For this purpose we factorize the a priori probability $p(\boldsymbol{w})$ of a word chain $\boldsymbol{w} = w_1 w_2 \dots w_m$ into the conditional probabilities of single words w_i having seen the predecessor words $w_1, \dots, w_{i-1}$. Generally the history for these n-gram models, i.e. the number of considered predecessing words, is limited to $n = 2$ (bigram) or $n = 3$ (trigram). Then the following equation holds:

$$p(\boldsymbol{w}) = p(w_1) \cdot \prod_{k=2}^{m} p(w_k | w_{k-n+1} \dots w_{k-1})$$

This approach for modeling languages can be further modified into category based n-grams, where the words from the lexicon are clustered together in categories or word classes. If we use unambiguous categories so that each word w_i belongs to one category z_i, we can estimate the probability for a word chain $\boldsymbol{w}$ by

$$p(\boldsymbol{w}) = p(z_1) \cdot p(w_1 \mid z_1) \cdot \prod_{k=2}^{m} p(z_k \mid z_{k-n+1} \dots z_{k-1}) \cdot p(w_k \mid z_k)$$

The idea beyond the polygrams is not to use a fixed length for the words belonging to the history but to use as much context as possible, that means we train n-grams up to a length where a robust training is still possible. Then we combine the different n-grams with the following interpolation:

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

L denotes the number of words in the lexicon, therefore the factor $\frac{1}{L}$ describes the so called zerogram. The interpolation weights $\rho_i(m)$ depend on the length of the actual word chain and can be estimated automatically with a cross validation technique.

3 Polygram Approach to Linguistic Analysis

As mentioned in Section 1 we reduce linguistic analysis to the extraction of pairs of semantic attributes and their corresponding value. In our domain of train-time-table informations, relevant attributes are e.g. *sourcecity, goalcity, time, date*. A detailed analysis of data collected in a field test [1] revealed that semantic attributes are expressed in only a few different syntactic constructs and we are convinced, that language models should be able to learn the differences between the constructs used for different attributes. Therefore we train an attribute–dependent language model for each of the attributes to be detected. During analysis, we decide which attribute is present in the utterance depending on the scores of the different language models.

A problem which we have to face is that generally a word chain contains more than only one semantic attribute. If we concentrate only on that one with highest probability a lot of semantic information will get lost. A first possibility to overcome this problem is to decide for each attribute based on stochastic models whether it is in the word chain or not. Therefore we train a model that scores the appearance of the attribute and one that scores the non-appearance. As before, the higher probability decides. Using this approach for each new word chain and for each of the semantic attributes we have to compute two scores and compare them.

A second possibility is to train models on combinations of semantic attributes. In the training data there is a limited amount of possible combinations of semantic attributes. E.g. if there are enough word chains containing *sourcecity* as well as *goalcity* we can train a specialized model on the appearance of this attribute combination.

4 Experiments & Results

For the experiments on shallow linguistic analysis using polygram language models we use the corpus described in [1] which is collected using the train-time-table information system EVAR. For training and test purposes we use a set of 10114 sentences. For these sentences we have a semantic annotation which we use as a reference for the detection of semantic attributes. Based on this annotation, the training corpus is split into several parts either in two parts, one with sentences containing a special attribute and a second one not containing it or in parts depending on the attribute combinations.

4.1 Experiment I

The semantic annotation was clustered together in three classes of semantic attributes namely *CITY*, *TIME* and *DATE*. For each attribute two models were trained, one for the appearance of the attribute in the word chain and one for the absence and this was done for the spoken word chain as well as for the recognized. The recognized word chains were also split up using the reference

annotation for the semantic based on the transliteration. Therefore a higher error rate is expected. The resulting data were split 2/3 to 1/3 for training and test (cf. Table 1). Since a word sequence from the test set might have been used by a different speaker from the training set (i.e. if the system asks *where do you want to leave from* most users answered *from Erlangen* or *from Nuremberg*), the column 'test $\neq$ train' gives the number of test sentences that were not observed during training.

				test $\neq$ train		spoken word chain			recog. word chain		
	total	train	test	spoken	recog.	YES	NO	RR	YES	NO	RR
CITY	3538	2359	1179	629	831	**99.6**	0.4	**91.0**	**84.8**	15.2	**82.0**
NOCITY	6576	4384	2192	581	795	13.7	**86.3**		19.5	**80.5**	
DATE	1870	1247	623	326	412	**94.7**	5.3	**90.1**	**65.3**	34.7	**84.0**
NODATE	8244	5496	2748	867	1205	11.0	**89.0**		11.7	**88.3**	
TIME	1742	1162	580	325	388	**94.1**	5.9	**90.6**	**64.5**	35.5	**85.3**
NOTIME	8272	5582	2790	874	1227	10.2	**89.8**		10.3	**89.7**	

Table 1. Number of word chains for training and test; Detection rates with the spoken (left) and recognized (right) word chain

4.2 Experiment II

In a second experiment we examine a more detailed classification. We now use three classes for each attribute, one class for the non-appearance of the attribute (NO), one for the unique appearance of it (ONLY) and one for describing that the attribute and other semantically relevant parts are in the word chain (PLUS). The data were split up in three parts for each attribute (cf. Table 2) again for the spoken and the recognized word chain.

The results for this experiment are reported in Table 3. The recognition rates show that with the polygram approach it is possible to distinguish between the single appearance of a certain attribute within a word chain and the appearance of the attribute along with others. This classification can then further be used in a hierarchical system for the detection of the semantic in a word chain.

5 Conclusion & Further Work

The experiments and results in Section 4 show that the polygram approach for shallow linguistic analysis is successful. The probabilistic models are able to learn automatically to distinguish between the semantic attributes and their realizations in word chains.

	total	train	test	test ≠ train spoken	test ≠ train recognized
CITYONLY	1668	1112	556	145	281
CITYPLUS	1870	1247	623	480	548
NOCITY	6576	4384	2192	581	795
DATEONLY	624	416	208	30	68
DATEPLUS	1246	831	415	305	349
NODATE	8244	5496	2748	867	1205
TIMEONLY	137	92	45	16	17
TIMEPLUS	1605	1070	535	313	375
NOTIME	8272	5582	2790	874	1227

Table 2. Number of word chains for training and test

	spoken word chain				recognized word chain			
	ONLY	PLUS	NO	RR	ONLY	PLUS	NO	RR
CITYONLY	**82.4**	17.4	0.2		**51.8**	28.8	19.4	
CITYPLUS	9.6	**90.0**	0.4	**85.5**	11.0	**78.5**	10.5	**73.8**
NOCITY	5.7	9.1	**85.2**		7.5	14.4	**78.1**	
DATEONLY	**92.3**	7.7	0.0		**72.1**	10.6	17.3	
DATEPLUS	11.6	**83.6**	4.8	**85.1**	5.6	**60.0**	34.4	**79.2**
NODATE	6.1	9.1	**84.8**		4.6	12.7	**82.7**	
TIMEONLY	**91.1**	8.9	0.0		**71.1**	11.1	17.8	
TIMEPLUS	8.8	**88.8**	2.4	**85.3**	6.0	**65.6**	28.4	**81.1**
NOTIME	7.2	8.2	**84.6**		6.9	8.8	**84.3**	

Table 3. Detection rates with the spoken (left) and recognized (right) word chain

The next step to evaluate this approach is to make further experiments with more semantic attributes to detect and to build models for the appearance of attribute combinations e.g. *sourcecity* and *goalcity* in one word chain, which is a combination senn quite often. Therefore the training data have to be examined, which combinations occur and which of them appear frequent enough to train robust language models depending on the semantic content.

Another direction for further research is the step towards a semantic segmentation. As yet reported in [2] the semantic attributes are realized using small syntactic constituents which should be localized in the word chain. After the localization the extraction of the corresponding value (cf. example in Section 1 with attribute *goalcity* and value *Hamburg*) is easier to perform than to get it out of the complete word chain.

6 Acknowledgement

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.

References

1. W. Eckert, E. Nöth, H. Niemann, and E.G. Schukat-Talamazzini. Real Users Behave Weird — Experiences made collecting large Human–Machine–Dialog Corpora. In Paul Dalsgaard, Lars Bo Larsen, Louis Boves, and Ib Thomsen, editors, *Proc. of the ESCA Tutorial and Research Workshop on Spoken Dialogue Systems*, pages 193–196, Vigsø, Denmark, June 1995.
2. 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.
3. E. G. Schukat-Talamazzini. *Automatische Spracherkennung – Grundlagen, statistische Modelle und effiziente Algorithmen.* Künstliche Intelligenz. Vieweg, Braunschweig, 1995.
4. H. Stahl, J. Müller, and M. Lang. An efficient top–down parsing algorithm for understanding speech by using stochastic syntactic and semantic models. In *Proc. of International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 397–400, Atlanta, USA, 1996.