COMBINING STOCHASTIC AND LINGUISTIC LANGUAGE MODELS FOR RECOGNITION OF SPONTANEOUS SPEECH

Wieland Eckert *Florian Gallwitz* *Heinrich Niemann*

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

ABSTRACT

In this paper we present a new approach of combining stochastic language models and traditional linguistic models to enhance the performance of our spontaneous speech recognizer. We compile arbitrary large linguistic context dependencies into a category based bigram model which allows us to use a standard beam-search driven forward Viterbi algorithm for real time decoding. Since this recognizer is used in a dialog system, the information about the last system utterance is used to build dialogstep dependent language models. This setup is verified and tested on our corpus of spontaneous speech utterances collected with our dialog system. Experimental results show a significant reduction of word error rate.

1. INTRODUCTION

In the last years it has been shown that the consideration of language constraints is vital for effective and efficient speech recognition. Typically, these language constraints are modeled in a so called *language model* which will restrict the allowed seqences of words in an utterance [7]. The *a priori* probability $P(\underline{w})$ for a word sequence $\underline{w} = w_1 w_2 \dots w_m$ can be expressed as a product of conditional probabilities $P(w_t|w_1 w_2 \dots w_{t-1})$. Approximation of the *history* of the word w_t is done by limiting the number of considered preceding words to n. For this n-gram approach n is typically restricted to $n = 2$ (bigram) or $n = 3$ (trigram):

$$P(\underline{w}) = P(w_1) \cdot \prod_{t=2}^{m} P(w_t|\underbrace{w_{t-n+1} \dots w_{t-1}}_{n-1})$$

Another type of stochastic language models are category based n-gram models [3]. Words are pooled in categories or word classes, usually under linguistic aspects. If one word is allowed to belong to more than one category, all possible category sequences $\underline{z} = z_1 z_2 \dots z_m$ leading to a word sequence $\underline{w} = w_1 w_2 \dots w_m$ have to be considered when calculating it's probability:

$$P(\underline{w}) = \sum_{\underline{z}} P(z_1)P(w_1|z_1) \cdot \cdot \prod_{i=2}^{m} P(z_i|\underbrace{z_{i-n+1} \dots z_{i-1}}_{n-1})P(w_i|z_i)$$

Unfortunately, the search space of a Viterbi continous speech decoder grows exponentially with the order n of the language model. Thus, for large vocabulary real time Viterbi decoding on standard hardware, the context has to be reduced to $n = 2$.

On the other hand, linguistic models can easily describe large context dependencies using a *grammar* G to generate a *language* $L(G)$ of accepted word sequences. Grammars can be used as language models for speech recognition [2]. The approach is quite similar to the usage of stochastic language models. The conditional probabilites $P(\underline{w})$ of allowed sequences can be made by following paths in the generation of $L(G)$ and multiplying the inverse branching factor at each step. Grammar based models are known to be very restrictive and have a quite low perplexity for a comparable coverage. Unfortunately their robustness against spontaneous speech phenomena is fairly limited.

Thus, it seems promising to combine both types of models: linguistic models are expected to lead to a better word accuracy for "clean" utterances whereas stochastic models are much more robust for spontaneous speech.

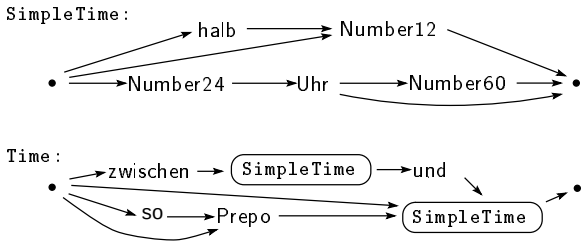
An approach to represent a finite state grammar as a word bigram for recognition of strongly structured commands can be found in [8]. The lexicon size is considerably increased by indexing the words to maintain context information. It is shown how the overhead can be reduced drastically by using a tree structured lexicon. Our approach is based on a category based decoding algorithm. Thus, no lexicon entries have to be duplicated.

In this paper we present a new approach of combining linguistic models and stochastic models. First we describe the basic models for recognizing short phrases and combine the grammatical units to generate a linguistic bigram model. Then we show in detail the mechanism for combination of stochastic and linguistic models. Performance measures are evaluated using our corpus of spontaneous speech data collected by our spoken dialog system [6, 10] which is able to answer inquiries about German Intercity train connections. It is accessible via public telephone line since January 1994 and we are recording all calls to the system. Different phases of system performance are described in [5] as well as phenomena observed in spontaneous speech dialogs.

2. LINGUISTIC BIGRAM MODELS

The linguistic models used in our approach can be represented as a finite state grammar or, graphically, as transition networks. They are constructed manually while investigating a subset of our collected corpus. They are not expected to cover *all* utterances of this subset but to represent frequent sentences such as:

> *Ich würde gerne morgen früh so gegen halb sieben von München nach Hamburg fahren* (I would like



halb:	halb, viertel, dreiviertel
so:	so
und:	und
zwischen:	zwischen
Number12:	eins, zwei, drei, vier, fünf, sechs, ... , zwölf
Number24:	null, ein, zwei, drei, vier, ... ,vierundzwanzig
Number60:	eins, zwei, drei, vier, ... , neunundfünfzig
Prepo:	ab, gegen, nach, um, vor
Uhr:	Uhr

Figure 1. Example of the transition network for German time expressions

> to take a train from Munich to Hamburg tomorrow morning about half past six)
> *ich möchte vor vier Uhr in Stuttgart ankommen* (I want to arrive in Stuttgart before 4 o'clock)
> *nein, am Freitag* (no, on Friday)

The first step of building these models is to define a set of not necessarily distinct word categories such as City, Number24, or Number60, which are the terminal symbols of the finite state grammar. They are used to define transition networks of arbitrary complexity, which are the nonterminal symbols of the finite state grammar. First we define transition networks for simple expressions, such as `SimpleTime` (Figure 1). These networks are used as building blocks to define networks for more complex expressions like `Time`. `Time` can handle German time expressions like:

> *zwischen sechs Uhr und sieben Uhr dreißig* (between six o'clock and seven thirty)

This process is continued to build networks that cover complete utterances, e.g. answers to the question "At what time would you like to leave?". A typical answer to this question is the eliptical utterance shown above, but of course complete sentences are possible, too:

> *Ich möchte zwischen sechs Uhr und sieben Uhr dreißig abfahren* (I would like to leave between six o'clock and seven thirty)

This kind of transition network is a model for one particular dialog step. It can be used as a dialogstep dependent model, assuming the recognizer in a dialog system is informed about the previous system utterance. Additionally, all dialogstep dependent networks can be combined to build one dialogstep independent transition network which does *not* depend on *a priori* information.

Before this kind of model can be stored in a category based bigram, every node of the transition network has to be a word category. Thus, we expand the models by successively inserting the simple networks into the more complex networks to build a flat transition graph. The nodes of every inserted subnetwork are marked to ensure that the nodes of the resulting graph are distinct. For example, when inserting the subnetwork `SimpleTime` into `Time` (Figure 1) in two different positions the resulting `Time` network contains two *different* nodes for Uhr.

In our case, 439 different words are stored in 40 categories and 8 subnetworks are used to build 5 dialogstep dependent networks and one dialogstep independent network. After expansion, the dialogstep independent transition graph contains 231 nodes. Each node represents one of the 40 word categories and is identified by a unique name. These 231 nodes are used as categories for our category based bigram. Therefore all relevant history information can be preserved by storing only the predecessor category during decoding. The conditional emission probabilities $P(w_i|z_i)$ of word w_i in category z_i can either be assigned uniformly or they can be estimated by parsing the utterances of our corpus that are covered by the linguistic models. Currently, the bigram transition probabilities $P(z_j|z_i)$ are assigned uniformly according to the branching factor. The resulting category based bigram model is suitable for direct usage within the recognizer [10].

3. COMBINATION OF LANGUAGE MODELS

Construction of linguistic models aims to incorporate as much of the utterance history as possible into the recognition process. Many different grammars and formalisms have already been built and used in natural language recognition and understanding. Unfortunately there is not great success in building models for spontaeous speech. Typical effects of spontaneous speech are (by definition) spontaneous and are usually not conforming to any grammars. On the other hand, stochastic n-gram models are robust against effects of spontaneous speech,

It is well known, that a more specialized model results in better recognition rates, but robustness is only gained using more general models. The basic idea of our approach is to combine a highly specialized linguistic model and a robust stochastic model. Combination of language models is done the same way as combining HMMs: the resultung model is made up by parallel search through both of them. It is up to the Viterbi recognizer to find a path through one of the models. Since several paths are possible, the recognizer will decide on the path with the highest probability. Paths within the linguistic language model and within the stochastic model are treated equally. Therefore the combined model is expected to inherit the specific advantages of both components: the highly specialized linguistic model is expected to lead to better paths for grammatical correct utterances whereas the stochastic model ist expected to cover the spontaneous effects of speech.

Both models are represented as categorial bigrams. In order to build a parallel model the underlying HMMs have to be combined. An HMM $\mathcal{M} = (\underline{A}, \underline{B}, \underline{\pi})$ consists of transition probabilities $\underline{A}$, emission probabilities $\underline{B}$ and a vector $\underline{\pi}$ of probabilities of initial states. Combining two HMMs $\mathcal{M}_1$ and $\mathcal{M}_2$ is performed by building a new HMM $\mathcal{M}$ according to the following block matrices:

$$\underline{A} = \left(\begin{array}{c|c} \underline{A}_1 & 0 \\ \hline 0 & \underline{A}_2 \end{array}\right), \underline{B} = \left(\begin{array}{c|c} \underline{B}_1 & 0 \\ \hline 0 & \underline{B}_2 \end{array}\right), \underline{\pi} = \begin{pmatrix} 0.5 \cdot \underline{\pi}_1 \\ 0.5 \cdot \underline{\pi}_2 \end{pmatrix}$$

Using this method, the transition from one HMM to the other one is prohibited and the inital states of both models are entered with equal probability. Therefore, it is possible to combine arbitrary language models but to keep the paths separated. All paths stay in the same language model, there is no transition from one model to the other.

sample	calls	user turns	words
training	804	7732	27852
validation	54	441	1577
test	234	2383	8346

Table 1. Overview of training-, validation-, and test sample

4. DIALOGSTEP DEPENDENT MODELS

In a spoken dialog system the interaction between user and system leads to a quite predictable kind of utterances. In fact, the system uses predictions to restrict the recognizers search space for subsequent user utterances. This is performed by using different, dialogstep dependent language models. One observation is that the system has a set of typical questions, e.g. asking for the departure city. An average user usually answers these questions. To evaluate this, the corpus was partitioned into sets of user utterances according to the previous system utterance. Thus, the actual utterance had no influence on the dialogstep it was assigned to. That way, we got a subcorpus of user utterances for each of our 14 dialogsteps. Using these subcorpora we made up dialogstep dependent stochastic language models.

When building dialogstep dependent partitions of the whole corpus, a subcorpus might have insufficient size. Some of the dialogsteps have quite low occurance. Small subcorpora of training data would lead to a mismatch between model and reality. We solved this problem by generalizing the dialogsteps. Generalization is done by combining dialogsteps which lead to similar user utterances. The resulting subcorpus is considerably larger and sufficient for the training of stochastic models.

As shown in the previous section, the usage of linguistic models promises better performance. Therefore we made up linguistic models for each dialogstep, too [1]. The dialogstep dependent linguistic models only cover uttererances that correspond to the preceding system utterance and are quite strict. Unfortunately, in a dialog system the user is free to deviate from the modeled behavior. As a consequence such an utterance would be recognized poorly: it is out of the coverage of that linguistic model.

Again, the solution is to combine a highly specialized, dialogstep dependent linguistic model and a more general (dialogstep dependent) stochastic model as described in the previous section. Both of them are better suited for a particular dialogstep since they are based on a specific subcorpus. Therefore they are expected to result in better recognition rates as well as faster computation.

5. EXPERIMENTS & RESULTS

Our collection of spontaneous speech data, which totals to 8 h 36 min of speech signals, is divided into a training sample used for training of acoustic parameters and stochastic language models, a validation sample for optimizing recognizer parameters, and a test sample (Table 1). Varying acoustic conditions and system development steps are represented in these samples proportionally (Table 2). The linguistic models were built manually while investigating 1742 utterances that were recorded by microphone (Table 2) *before* we defined our training sample. For comparison, we removed the 536 microphone utterances from our test sample for all experiments reported in this paper.

All linguistic models were built without training of transition probabilities and word emission probabilities; these were assigned uniformly. The dialogstep dependent linguistic

users	acoust. cond.	turns
seminaive	microphone	1742
seminaive	PABX	584
experts	PABX	491
naive,seminaive	PSTN	7739

Table 2. Overview of user type and acoustic conditions

model was built using the five dialogstep dependent transition networks 'INITIAL', 'TIME', 'GOALCITY', 'SOURCECITY', 'DATE' together with the dialogstep *in*dependent transition network as backdrop model. The dialogstep independent model covers 67.1 percent of the utterances in the test sample, it's perplexity on this subset is 20.34. The dialogstep dependent linguistic model covers 64.0 percent of the test sample, its perplexity on these utterances is 15.26.

The stochastic models are trained using a set of 300 distinct categories, which consist of 64 handcrafted categories, 235 single-word categories containing the most frequent words not included in the handcrafted categories and one category for all other words. The lexicon consists of 1558 words.

Seven different dialogstep dependent stochastic models were trained on distinct subsets of the training sample and tested on the corresponding subsets of the test sample, whereas the general stochastic model was trained on the whole training sample. For example, the 'TIME'-model was trained and tested on utterances following questions like "At what time would you like to leave?" or "At what time would you like to arrive?".

Our approach of combining bigrams described in section 3 is suitable for combining arbitrary category based bigrams. As we were particularly interested in combining stochastic and linguistic models and we had two stochastic and two linguistic models, there were four possible combinations. Additionaly we evaluated our system with all models for themselves and with no language model at all.

The perplexities of all stochastic and combined linguistic-stochastic models on the test sample can be found in Table 3. As we expected, there was no reduction in perplexity by combining different models, since the probability of all utterances that are not covered by our linguistic model is halved. The resulting word accuracies of all models on the test sample can be found in Table 4. The dialogstep dependent linguistic model performs much poorer than the dialogstep independent linguistic model since it is very restricted. When combined with a stochastic model, the dialogstep dependent linguistic model outperforms the dialogstep independent linguistic model. Combining the baseline system (dialogstep independent stochastic model) with a dialogstep dependent linguistic model reduces the word error rate by 3.3 percent while a dialogstep dependent stochastic model reduces the word error rate by 4.3 percent. A 6.0 percent reduction is achieved by combining dialogstep dependent linguistic and stochastic models.

The corresponding real time factors on a HP735 workstation can be seen in Table 5. Of course, the dialogstep dependent linguistic model is much faster than any other model (70 percent CPU-time reduction compared to the baseline system). The dialogstep dependent stochastic model leads to a 15 percent reduction of CPU-time. The best performing model, the combination of dialogstep dependent models, does not need significantly more CPU-time than the baseline system.

linguistic model		stochastic model	
		general	dst.dep.
	none	22.00	18.22
	general	24.47	20.20
	dst.dep.	23.36	18.76

Table 3. Perplexities of different combinations of language models

linguistic model		stochastic model		
		none	general	dst.dep.
	none	47.69	26.87	25.72
	general	38.25	26.36	25.67
	dst.dep.	46.49	25.99	25.26

Table 4. Word error rates resulting from different combinations of language models

6. SUMMARY

In this paper we presented a uniform approach to combine linguistic and stochastic language models. Liguistic language models define a recognition grammar and cover large context dependencies, but are very restricted. On the other hand, stochastic models are more robust. Linguistic models can be represented in the same formalism as stochastic models. We use a categorial bigram representation for both of them. For combining two (arbitrary) language models we constructed a parallel model which allows the unmodified recognizer to search in both models in parallel. Preliminary experiments with a reduced training set have been conducted and showed a substantial improvement in recognition rate.

A sample of about 7 700 utterances was used for training stochastic dialogstep dependent language models. A smaller subset of about 1 700 utterances was selected for constructing the linguistic models. The new combined model reduces the word error rate by 3.3 percent and leads to a marginal increase in computation time. While this is much less than our preliminary experiments with reduced training sets promised, it is still a remarkable improvement for recognition of spontaneous speech. We think that our approach is of special interest for domains where user utterances are quite restricted and only small training samples are available.

7. FURTHER WORK

Currently we use uniform distributions for the linguistic models. These should be replaced by estimations according to training set. The transition from one language model to another is prohibited. More complex utterances could be handled by allowing transitions between different language models in special states, e.g. during silence periods. For further evaluation, calculation of the word accuracy (WA) should be substituted by calculation of the accuracy of semantic concepts (SA). When integrated in a spoken dialog system, the recognizers SA plays a dominant role while it's WA is irrelevant. The role of language models will be further investigated.

ACKNOWLEDGEMENTS

The work presented in this paper was partly supported by the DFG (German Research Foundation) under contract number 810 830-0. We would also like to thank all colleagues who were involved in the installation of our dialog system.

linguistic model		stochastic model		
		none	general	dst.dep.
	none	3.98	3.08	2.60
	general	2.04	3.86	3.50
	dst.dep.	0.64	3.53	3.10

Table 5. Real time factor resulting from different combinations of language models

REFERENCES

[1] U. Ackermann. Bestimmung von dialogschrittabhängigen Sprachmodellen aus Dialogkorpora. Technical report, Studienarbeit, Lehrstuhl für Mustererkennung (Informatik 5), Universität Erlangen–Nürnberg, 1994.

[2] L. Bahl, J. Baker, P. Cohen, A. Cole, F. Jelinek, B. Lewis, and R. Mercer. Automatic Recognition of Continuously Spoken Sentences from a Finite State Grammar. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, pages 418–421, Tulsa, 1978.

[3] A.-M. Derouault and B. Merialdo. Natural Language Modeling for Phoneme-to-Text Transcription. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(6):742–749, 1986.

[4] P. Dumouchel, V. Gupta, M. Lennig, and P. Mermelstein. Three Probabilistic Language Models for a Large-Vocabulary Speech Recognizer. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, pages 513–516, New York, 1988.

[5] W. Eckert, E. Nöth, H. Niemann, and E.-G. Schukat-Talamazzini. Real Users Behave Weird — Experiences made collecting large Human–Machine–Dialog Corpora. In P. Dalsgaard, L. B. Larsen, L. Boves, and I. Thomsen, editors, *Proc. of the ESCA Tutorial and Research Workshop on Spoken Dialogue Systems*, pages 193–196, Vigsø, Denmark, June 1995.

[6] W. Eckert, T.Kuhn, H. Niemann, S. Rieck, A. Scheuer, and E. G. Schukat-Talamazzini. A Spoken Dialogue System for German Intercity Train Timetable Inquiries. In *Proc. European Conf. on Speech Communication and Technology*, pages 1871–1874, Berlin, Germany, Sept. 1993.

[7] F. Jelinek, R. Mercer, and L. Bahl. Continuous Speech Recognition. In P. Krishnaiah and L. Kanal, editors, *Handbook of Statistics*, volume 2, pages 549–573. North-Holland, 1982.

[8] U. Kilian, F. Class, A. Kaltenmeier, and P. Regel–Brietzmann. Representation of a Finite State Grammar as a Bigram Language Model for Continuous Speech Recognition. In *Proc. European Conf. on Speech Communication and Technology*, pages 1241–1244, Madrid, Spain, Sept. 1995.

[9] A. Paeseler and H. Ney. Continuous Speech Recognition Using a Stochastic Language Model. In *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, pages 719–721, Glasgow, 1989.

[10] E. Schukat-Talamazzini, T. Kuhn, and H. Niemann. Speech Recognition for Spoken Dialog Systems. In H. Niemann, R. De Mori, and G. Hanrieder, editors, *Progress and Prospects of Speech Research and Technology*, number 1 in Proceedings in Artificial Intelligence, pages 110–120. Infix, 1994.