ERGODIC HIDDEN MARKOV MODELS AND POLYGRAMS FOR LANGUAGE MODELING *

T. Kuhn *H. Niemann* *E.G. Schukat-Talamazzini*
Universität Erlangen-Nürnberg, Lehrstuhl für Mustererkennung (Informatik 5)
Martensstraße 3, 91058 Erlangen, F.R. of Germany
E-mail: kuhn@informatik.uni-erlangen.de

ABSTRACT

In this paper we present two new techniques for language modeling in speech recognition. The first technique is based on ergodic discrete density Hidden Markov Models (HMM) which can be applied to bigrams based on word categories. This statistical approach of the so-called *Markov bigrams* enables an efficient unsupervised learning procedure for the bigram probabilities with the well-known Baum-Welch algorithm. Furthermore, maximizing the model-conditional probability is equivalent to minimizing the perplexity of the training corpus. The second technique is based on *polygrams* which are an extension of the bigram ($n = 2$) or trigram ($n = 3$) grammars to any possible value of n. According to the smoothing techniques for bigram or trigram models, the probabilities of the n-grams in the polygram model are interpolated using the relative frequencies of all n'-grams with $n' \leq n$. Both techniques were evaluated on the ATIS corpus by computing the test set perplexity. Furthermore we integrated the *Markov bigrams* as well as the *polygrams* into our word recognizer for continuous speech. Experimental results on a German database are discussed using the *N-best paradigm* to reorder the generated word sequences according to the sentence probability of the language model.

1. INTRODUCTION

It has been shown in the past years that the consideration of linguistic constraints by language models during the recognition process is very important to achieve a good system performance. The language model provides information to guide the recognizer through the search space by discarding unlikely word sequences. Typically, the linguistic constraints are modeled by statistical language models where the *a priori* probability $P(\underline{w})$ of a word sequence $\underline{w} = w_1 w_2 \ldots w_m$ is computed [2].

Let $\mathcal{V} = \{W_1, W_2, \ldots, W_L\}$ be a vocabulary of L words. The *a priori* probability $P(\underline{w})$ for the word sequence $\underline{w} = w_1 w_2 \ldots w_m$ with $w_i \in \mathcal{V}$ can be expressed as a product of the conditional probabilities $P(w_t|w_1 w_2 \ldots w_{t-1})$:

$$P(\underline{w}) = P(w_1) \cdot \prod_{t=2}^{m} P(w_t|\underbrace{w_1 w_2 \ldots w_{t-1}}_{\text{history } \mathfrak{H}}) \qquad (1)$$

The sequence $w_1 w_2 \ldots w_{t-1}$ is called the history of the underlying stochastic process for $P(\underline{w})$. The probability $P(\underline{w})$ can be approximated by restricting the history to the preceding $n-1$ words, which leads to the concept of n-gram models, with:

$$P(w_1 w_2 \ldots w_m) = P(w_1) \cdot \prod_{t=2}^{m} P(w_t|\underbrace{w_{t-n+1} \ldots w_{t-1}}_{(n-1)}) \qquad (2)$$

Usually, the n-gram probabilities $P(w_t|w_{t-n+1} \ldots w_{t-1})$ are estimated by the relative frequencies according to the formula:

$$\hat{P}(w_t \mid w_{t-n+1} \ldots w_{t-1}) = \frac{\#(w_{t-n+1} \ldots w_t)}{\sum_{v \in \mathcal{V}} \#(w_{t-n+1} \ldots w_{t-1} v)} \qquad (3)$$

where $\#(.)$ denotes the frequency of a certain n-gram. The more context is considered, the larger the training corpus has to be to guarantee a robust parameter estimation of the L^n n-gram probabilities. Even if a huge training corpus is available and the history is restricted to one or two preceding words, there will be a large amount of possible bi- or trigrams which will never occur in the training data. As a consequence, the probabilities of these n-grams would be zero, which can be embarrassing in the recognition process.

A solution to this problem is given by two different methods. One approach is to explicitly reduce the parameter space by building equivalence classes where each word belongs to one or more classes [9]. Another approach is to increase the robustness of the estimated conditional n-gram probabilities $P(w_t|w_{t-n+1} \ldots w_{t-1})$ by backing-off the statistics of unseen events [7], by linear interpolation of lower-order models [4], or by co-occurrence smoothing [5].

In the first method, the probability for the observation of the word sequence $\underline{w}$ can be expressed by the following equation:

$$P(\underline{w}) = \sum_{\underline{c} \in \mathcal{C}^m} \prod_{t=1}^{m} P(c_t|c_1 c_2 \ldots c_{t-1}) \cdot P(w_t|c_t) \qquad (4)$$

$\mathcal{C} = \{C_1, C_2, \ldots, C_K\}$ is a set of K word categories or parts of speech (POS) and $\underline{c} = c_1 c_2 \ldots c_m$ denotes a sequence of word categories according to the word chain $\underline{w}$. In the equation above, we assumed that the membership of word w_t is independent of the membership of the categories of the preceding words. Restricting the history to the last word

*This work was partly funded by the Commission of the European Union under ESPRIT contract P 2218 (*SUNDIAL*) as well as by the German Ministry for Research and Technology (*BMFT*) within the project KAN INF 18. Only the authors are responsible for the contents.

leads to the special case of a bigram model:

$$P(\underline{w}) = \sum_{c \in C^m} \prod_{t=1}^{m} P(w_t|c_t) \cdot P(c_t|c_{t-1}) \qquad (5)$$

In comparison to the word-based n-grams the main advantage of the category model is based on the fact that the parameter space can be reduced drastically, because typically the number of categories is significantly lower than the number of words. On the other hand, there are two drawbacks of the category model. First, a set of categories has to be defined and second, the training corpus has to be tagged in advance. Both problems can be solved automatically using agglomerative clustering methods [8], iterative Viterbi alignment [12], or the Baum-Welch re-estimation procedure [11, 3].

In this paper we present two new methods for statistical language modeling. The concept of the *Markov bigrams* enables an unsupervised learning procedure of a bigram model based on word categories using an ergodic discrete density HMM. The *polygrams* are an extension of the well-known bi- or trigrams where a history of more than two words can be considered. This concept enables us to capture longer context information as in phrases like *"Show me the flights from ..."*. Also in the word *polygram* approach, word categories can be used to reduce the parameter space. Both methods will be evaluated on the ATIS MADCOW [10] corpus with respect to the test set perplexity. Furthermore, we integrated both language models into a continuous speech recognizer and we will present experimental results for a German speech database.

2. MARKOV BIGRAMS

Assuming no deterministic tagging of word sequences in word classes, the *a priori* probability $P(\underline{w})$ of a word sequence $\underline{w}$ is computed according to equation (5). Since this formula describes a first order *Markov process*, the bigram model can be represented as an ergodic discrete density HMM $\underline{\lambda} = (\underline{\Pi}, \underline{A}, \underline{B})$ with

$$\begin{aligned} a_{ij} &= P(c_t = C_j \mid c_{t-1} = C_i) \\ b_{jk} &= P(w_t = W_k \mid c_t = C_j) \\ \pi_i &= P(c_1 = C_i) \end{aligned}$$

The hidden states represent the possible categories and the output of the HMM $\underline{\lambda}$ represents the words. The transition probabilities are summarized in the KxK matrix $A = [a_{ij}]$ where a_{ij} defines the bigram probability $P(C_j|C_i)$ between category C_i and C_j. The conditional word probabilities $P(W_k|C_j)$ that word W_k belongs to category C_j is represented in a KxL matrix $\underline{B} = [b_{jk}]$. As a consequence each word may belong to each category.

Given the HMM parameters $\underline{\lambda} = (\underline{\Pi}, \underline{A}, \underline{B})$, the probability $P(\underline{w})$ of a word sequence $\underline{w}$ can be computed by the *forward* algorithm and the well-known *Baum-Welch* algorithm can be used for parameter training. Since the categories are treated as hidden states, the probabilities can be estimated in an unsupervised manner which means that no tagging of the training corpus is necessary. Only the number of different word classes has to be chosen in advance. According to the maximum likelihood criterion of the *Baum-Welch* algorithm, maximizing the model conditioned probability is equivalent to minimizing the perplexity

$$\varphi_x(w_1 w_2 \ldots w_m) = \frac{1}{\sqrt[m]{P(w_1 w_2 \ldots w_m)}}$$

of the training sequence $w_1 w_2 \ldots w_m$.

As it is well known, the complexity of the parameter training via the *Baum-Welch* algorithm is approximately $O(K^2 m)$. This is due to the fact, that the $\alpha_t(j)$ probabilities for $1 \leq t < m, 1 \leq j \leq K$ are computed by

$$\alpha_{t+1}(j) = \sum_{i=1}^{K} \alpha_t(i) a_{ij} b_{jw_t} \qquad (6)$$

The main effort of the formula above is the computation of the N products $a_{ij} b_{jw_t}$. Especially, using an ergodic model in which all transition probabilities a_{ij} are non-zero, results in large computational effort. We developed a method to reduce the complexity of the *Baum-Welch* algorithm significantly. Following the observation that only a few products in equation (6) exceed zero, we define for each category C_i and each word W_k a set of categories $\mathcal{Q}(i,k)$ by

$$\begin{aligned} \mathcal{Q}(i,k) &= \{j \mid a_{ij} \cdot b_{jk} \leq \theta \cdot \rho(i,k)\} \\ \text{with} \qquad \rho(i,k) &= \max_j \{a_{ij} \cdot b_{jk}\} \end{aligned}$$

θ defines a threshold with $0 < \theta < 1$. $\mathcal{Q}(i,k)$ summarizes all categories C_j with the highest transition probability from category C_i to C_j and emitting word W_k. In the following experiments we adjust θ in such a way that $|\mathcal{Q}(i,k)| = p$, e.g. $\theta = \theta(i,k)$. The revised computation of the α (and β) variables according to

$$\alpha_{t+1}(j) = \begin{cases} \sum_{i=1}^{N} \alpha_t(i) a_{ij} b_{jw_{t+1}} & j \in \mathcal{Q}(i, w_{t+1}) \\ 0 & j \notin \mathcal{Q}(i, w_{t+1}) \end{cases}$$

leads to a complexity of $O(pKm)$. A similar technique is used for the update procedure during training. Experiments on ATIS have shown that $p = 8$ is sufficient to accelerate the training procedure significantly without any increase of the perplexity.

3. POLYGRAMS

As mentioned in the introductory section it seems worthwhile to consider word histories of arbitrary size in order to capture even long-spanning statistical dependencies between words. Therefore we propose a method in which the conditional word probabilities on the right hand side of equation (1) are evaluated without artificially cutting down the word history $\mathfrak{H}$ to a prespecified maximum size as is the case in equation (2).

For that purpose the complete set of training data statistics has to be stored, which consists of the occurrence counts of all *polygrams* (i.e. unigrams, bigrams, trigrams, and so forth) observed at least once in the training material. From the *polygram* counts we compute maximum likelihood (ML) estimates $\hat{P}(w_n|w_1 \ldots w_{n-1})$ of the conditional n-gram probabilities using equation (3). This estimate, however, will obviously disappear if the accompanying *polygram* $w_1 \ldots w_n$ was absent in the training set. Thus, a smoothed distribution is substituted into the language model equation which is obtained as a linear combination of ML estimates of conditional probabilities with successively reduced word history:

$$\begin{aligned} \tilde{P}(w_n|w_1 \ldots w_{n-1}) &= \lambda_0 \cdot \frac{1}{L} + \lambda_1 \cdot \hat{P}(w_n) \\ &+ \sum_{i=2}^{n} \lambda_i \cdot \hat{P}(w_n|w_{n-i+1} \ldots w_{n-1}) \end{aligned} \qquad (7)$$

For a vocabulary of size L, the expression $\frac{1}{L}$ represents the *uniform* (or the *zero-gram*) distribution.

The interpolation coefficients $\lambda_0, \ldots, \lambda_n$ have to fulfill the condition $\sum \lambda_i = 1$. They are optimized through several EM iterations [6] performed on a cross-validation corpus which has been chosen different from the training set and the test set. For a more concise modeling, a functional dependence of the weights from the word history $\mathfrak{H}$ is introduced by:

$$\lambda_i \;=\; \lambda_i(\mathfrak{H}) \;=\; \lambda_i(\max\{\nu | \#(w_{n-\nu} \ldots w_{n-1}) > 0\})$$

Our policy simply examines whether the sequence of the last ν history words is observed in the training data; it yielded the best results so far. Any further specialization of the interpolation weights led to an overadaptation of the language model to the cross-validation data, whilst showing no improvement of the test perplexity.

A *polygram* model can be based on words as well as on *non-overlapping* word categories. The latter case is formalized by equation (4); note however that the summation becomes obsolete because word categories have been assumed unique. After mapping the word items of the training data to category labels, the conditional category *polygrams* are estimated by linear interpolation (equation (7)). The category-dependent word probabilities are computed from the occurrence counts using *Jeffrey's* formula:

$$\tilde{P}(w_n|c_n) \;=\; (\#(w_n) + 1) \;/\; \sum_{v \in c_n} (\#(v) + 1)$$

4. EXPERIMENTS ON ATIS

We performed experiments on the ATIS text corpus as part of a collaboration with the Centre de Recherche Informatique de Montréal (CRIM) within the German BMFT project KAN INF 18. For training we used the ATIS2 MADCOW corpus [10]. The development set for estimating the *polygram* weights consisted of the evaluation set of NOV92. The evaluation set of FEB92 was used for computing the test set perplexity φ_x.

For the experiments with *Markov bigrams* we varied the number of categories from 1 to 400 (see Table 1). The initialization of the parameters was done *randomly*. The output probabilities of unknown words in the training set was set to 10^{-3} for each category. The more categories are distinguished, the smaller are the perplexities φ_x. Using more than 400 categories did not result in smaller perplexities.

#cat	1	50	100	150	200	300	**400**
φ_x	181.9	36.0	25.2	22.3	21.2	21.0	**19.9**

Table 1. Perplexities for Markov bigrams on ATIS

For the experiments with word-based *polygrams* we varied the maximum length n of n-grams considered in the model (see Table 2). It appears that hexagrams ($n = 6$) are sufficient to capture the context information in phrases or idioms. The consideration of more context information did not result in smaller perplexities.

n	1	**2**	3	4	5	6	∞
φ_x	173.8	**23.3**	17.6	17.1	17.0	16.9	**16.9**

Table 2. Perplexities for word-based polygrams on ATIS

Putting only city names, months, weekdays, and different numeral classes in altogether nine single categories results in a small improvement with $\varphi_x = 16.6$ A comparison of the word-based *polygrams* and the category-based *Markov bigrams* indicates that the consideration of more context information results in lower perplexities. However, if the context of the *polygrams* is restricted to bigrams, the category-based approach of the *Markov bigrams* is superior. We achieved an improvement of about 15% from perplexity 23.9 to 19.9 using *Markov bigrams*. This is due to the fact, that for the *Markov bigrams* the context information is coded in the category set and the affiliation of a word to a category is extremely ambiguous (each word belongs to each category) which is expressed by the category-dependent word probabilities $P(W_i|C_i)$.

5. THE RECOGNITION SYSTEM

The speech signal is sampled at 16 kHz, quantized with 14 bit and partitioned into 10 msec frames. For each frame a 256 point FFT with non-overlapping windows is computed. The result of the feature extraction module is a vector in the $\Re^{24}$ consisting of 11 mel-cepstrum coefficients, the corresponding delta mel-cepstral coefficients, and one coefficient for the energy and delta energy. The derivatives are computed using linear regression in a 9 frame neighborhood. The principal phonetic subword unit of the semi continuous HMM based recognizer is the *polyphone* representing a generalized context–dependent subword unit surrounded by arbitrary context size. [13]. The context items may also include suprasegmental markers or even word boundaries. This ensures that large-scaled contextual effects are properly statistically modeled. Design of the models and training of the HMM parameters is performed by the ISADORA system [14]. In the baseline system, we modeled all words and polyphones (syllable markers are included in the contexts) if their number of occurrence exceeds a threshold of 50. Using a test vocabulary of 1081 words results in 2991 subword units and 8674 probability density functions. For the experiments, we used a vector quantizer with 220 classes which was initialized by merging 44 phone–specific Gaussian 5–mixtures. This codebook was re-estimated three times by semi-continuous Baum-Welch training. Language modeling was incorporated using a standard bigram based on 95 different word categories which were defined according to morphological, syntactic, and semantic characteristics [15]. Only a few words belong to exactly two categories.

For *training*, we used about 11 hours of speech data spoken by 31 female and 48 male speakers. Each of the speakers uttered a unique set of 100 different application dependent sentences in the discourse domain of time table inquiries for trains. The *development set* (DEV) consists of 400 utterances spoken by 1 female and 3 male speakers. Each of these speakers uttered the same corpus of 100 application dependent sentences. The test set (TEST) of about 1.5 hours of speech contains 1,400 different application dependent sentences spoken by 10 female and 10 male speakers. The sentences in the *test set* cover 4 different situational contexts whereas all sentences in the development set as well as in the training set represent *initial* dialog utterances from the train time table domain exclusively.

6. EXPERIMENTAL RESULTS

In this section, we present experimental results for our continuous speech word recognizer using the proposed language models in the *N-best paradigm* to reorder the generated word sequences according to the sentence probability of the language model. The *Markov bigrams* as well as the *polygrams* were trained on 2,027 sentences with 10,890 words.

The development set to determine the weights λ_i via cross validation consists of 100 sentences with 800 words. We investigated word-based as well as category-based *polygrams*. The category set for the *polygrams* consists of 129 different word categories which are an extension of the category definition of the standard bigram used in the recognizer during search. Words which are in more than one category were put in a single category. For the experiments with the *Markov bigrams*, we adjust the number of possible categories to $K = 95$ (as in the standard bigram) and the initialization of the parameters was done *randomly*.

After the N best sentences were generated, each word sequence $\underline{w}$ is rescored according to

$$score(\underline{w}) = P(\underline{o}|\underline{w}) \cdot P(\underline{w})^{\alpha} \cdot \varpi^{\#words} \qquad (8)$$

$P(\underline{o}|\underline{w})$ denotes the acoustic score, ϖ denotes the word penalty to adjust deletions and insertion and α terms a weight to balance the acoustical and linguistical score. α and ϖ were adjusted on the development set.

lg-model →	standard	w-poly	c-poly	markov
φ_x	108.5	85.7	53.9	84.4
WA	**86.0**	86.2	**89.6**	87.2
SA	47.7	46.0	55.8	49.3

Table 3. Perplexities for polygrams on ATIS

Table 3 summarizes the results achieved on the test set TEST. The word accuracy and sentence accuracy corresponds to the best scored word sequence according to equation (8). φ_x is the test set perplexity of the language model. Note, that the category-based *polygrams* (c-poly) are superior to the word-based *polygrams* (w-poly) which is based on the fact that the generalization of the category-based *polygrams* is better if only a small training corpus is available. In comparison to the output of the recognizer with the standard bigram the word accuracy could be increased from 86.0% to 89.6% or, equivalently, the word error rate was reduced by 26%. The use of the *Markov bigrams* (markov) only results in a small increase of the recognition performance which is due to the fact that the context is restricted to the preceding word.

Furthermore, it can be seen in Table 3, that a better perplexity φ_x on a test set will lead to a better recognition performance.

7. CONCLUSIONS

We presented two new techniques for language modeling. The *Markov bigrams* enable an unsupervised learning procedure using the *Baum Welch* algorithm to estimate a category-based bigram model. The categories are extremely ambiguous, because each word belongs to each category. Since the categories are treated as hidden states, the bigram parameters can be learned unsupervised. On the other hand, the ergodic HMM requires a large amount of computational effort. For parameter training, an acceleration algorithm was given which works quite well.

The *polygrams* are an extension of the well known bi- and trigrams by considering arbitrarily large context information. The training of polygram models is completed in relatively short time, and the competing language models (standard bigrams and *Markov bigrams*) were outperformed by *polygrams* with respect to test set perplexity as well as word accuracy. *Polygrams* can not only be used to rescore n-best sentence hypotheses but also for certain recognition task, e.g. the recognition of phrase boundaries [1].

REFERENCES

[1] A. Batliner and A. Kießling and U. Kilian and R. Kompe and H. Niemann and E. Nöth and P. Regel-Brietzmann. Automatic Classification of Prosodically Marked Phrase Boundaries in German. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, Adelaide, Australia, 1994.

[2] L.R. Bahl, F. Jelinek, and R.L. Mercer. A Maximum Likelihood Approach to Continuous Speech Recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 5(2):179–190, 1983.

[3] D. Cutting, J. Kupiec, J. Pedersen, and P. Sibun. A Practical Part-of-Speech Tagger. In *Proc. 3. Conf. on Applied Natural Language Processing*, pages 133–140, Trento, Italy, 1992. ACL.

[4] A.M. Derouault and B. Merialdo. Language Modeling at the Syntactic Level. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 1373–1375, San Diego, 1984.

[5] U. Essen and V. Steinbiss. Co-occurrence Smoothing for Stochastic Language Modeling. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 161–164, San Francisco, 1992.

[6] F. Jelinek and R.L. Mercer. Interpolated Estimation of Markov Source Parameters from Sparse Data. In E.S. Gelsema and L.N. Kanal, editors, *Pattern Recognition in Practice*, pages 381–397. North Holland, 1980.

[7] S.M. Katz. Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, 35(3):400–401, 1987.

[8] R. Kneser and H. Ney. Improved Clustering Techniques for Class-Based Statist ical Language Modelling. In *Proc. European Conf. on Speech Technology*, pages 973–976, 1993.

[9] R. Kuhn and R. De Mori. A Cache-Based Natural Language Model for Speech Recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(6):570–583, 1990.

[10] MADCOW. Multi Site Data Collection for a Spoken Language Corpus. In *Speech and Natural Language Workshop*. Morgan Kaufmann, 1992.

[11] B. Merialdo. Tagging Text with a Probabilistic Model. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 809–812, Toronto, 1991.

[12] H. Ney and U. Essen. On Smoothing Techniques for Bigram-Based Natural Language Modelling. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 825–828, Toronto, 1991.

[13] E.G. Schukat-Talamazzini, T. Kuhn, and H. Niemann. Das POLYPHON — eine neue Wortuntereinheit zur automatischen Spracherkennung. In *Fortschritte der Akustik (Proc. DAGA '93)*, pages 948–951, Frankfurt, 1993.

[14] E.G. Schukat-Talamazzini, H. Niemann, W. Eckert, T. Kuhn, and S. Rieck. Acoustic Modelling of Subword Units in the ISADORA Speech Recognizer. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 577–580, San Francisco, 1992.

[15] P. Witschel and G. Niedermair. Experiments in Dialogue Context Dependent Language Modelling. In G. Görz, editor, *KONVENS 92*, pages 395–399. Springer, Berlin, 1992.