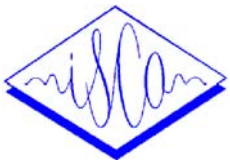

EUROSPEECH '97
5th European Conference on Speech Communication and Technology
Rhodes, Greece, September 22-25, 1997

ADAPTIVE TOPIC–DEPENDENT LANGUAGE MODELLING USING WORD–BASED VARIGRAMS

Sven C. Martin, Jörg Liermann, Hermann Ney
Lehrstuhl für Informatik VI, RWTH Aachen, University of Technology, D-52056 Aachen, Germany
E–mail: martin@informatik.rwth-aachen.de

ABSTRACT

This paper presents two extensions of the standard interpolated word trigram and cache model, namely the extension of the trigram model by useful word m–grams with $m > 3$ resulting into a varigram model, and the addition of topic–specific trigram models. We give the criteria for selecting useful m–grams and for partitioning the training corpus into topic–specific subcorpora. We apply both extensions, separately and in combination, to corpora of 4 and 39 million words taken from the Wall Street Journal Corpus and show that high reductions in perplexity of up to 19 % on the largest corpus are achieved. We also performed some recognition experiments.

1. INTRODUCTION

In this paper, a standard baseline language model is improved by an extension of the history length and by topic adaptation. The baseline model is the interpolation of two different language models: a word–based trigram model and a bigram cache model [4]. The parameters of the word–based trigram model are estimated using maximum likelihood. To smooth the word–based trigram model, we use absolute discounting with interpolation and singleton backing–off functions. The baseline model is extended as follows:

- The word–based trigram is replaced by a word–based variable–length m–gram, or *varigram* for short. In other words, the trigram model is extended by those word m–grams ($m > 3$) which are likely to reduce perplexity.
- The interpolation of the varigram model and the cache model is extended by a set of topic–specific word trigram models. All interpolation factors including the interpolation factors for the varigram model and the bigram cache model are dynamically adjusted according to the history using the EM–algorithm.

2. VARIGRAM MODEL

Extending a word trigram model by useful m–grams ($m > 3$) has been tried earlier, e.g. in [5] and [10]. In this approach, we use the concept of leaving–one–out (*L1O* for short) to select useful m–grams and compare it with a standard maximum likelihood selection scheme. We define the log–likelihood function on the training corpus

$$F = \sum_{h,w} N(h,w) \cdot \log q(w|h)$$

where $q(w|h)$ is either estimated by standard maximum likelihood

$$q(w|h) = \frac{N(h,w)}{N(h)}$$

or by L1O:

$$q(w|h) =$$

$$= \begin{cases} \dfrac{N(h,w)-1}{N(h)-1} - \delta(h) + N_+(h) \cdot \delta(h) \cdot q(w|\bar{h}) & \\ \qquad\qquad \textbf{if } N(h,w) > 1 \textbf{ and } N(h) > 1 & \\ (N_+(h)-1) \cdot \delta(h) \cdot q(w|\bar{h}) & \\ \qquad\qquad \textbf{if } N(h,w) = 1 \textbf{ and } N(h) > 1 & \\ q(w|\bar{h}) \qquad \textbf{if } N(h) = 1 & \end{cases}$$

with the number of seen distinct successor words of history h

$$N_+(h) := \sum_{w:\ N(h,w)>0} 1 \quad ,$$

$$\delta(h) := \frac{d_h}{N(h)-1}$$

as the probability mass discounted once from each event (h,w) seen in the training corpus, $\bar{h}$ as the generalized history of h (i.e. $h = (v,\bar{h})$), and d_h as the history–specific discounting value. In the unigram case, we use $N(w)$ instead of $N(h,w)$, the corpus size instead of $N(h)$, the overall number of seen distinct vocabulary words instead of $N_+(h)$, and $1/W$ instead of $q(w|\bar{h})$, with W as vocabulary size.

To select useful histories, we start with a set of *accepted* histories which consists of all word pairs h with $\sum_w N(h,w) > 0$, i.e. the histories of all distinct trigrams seen in the training corpus. We extend each accepted history h by each of its seen predecessor words v in the training corpus. The pairs (v,h)

form the set of *candidate* histories. We accept that candidate pair (v,h) which most increases the log–likelihood function:

$$\Delta F(v',h') = \sum_w N(v',h',w) \cdot \log \frac{q(w|v',h')}{q(w|h')}$$

$$(v,h) = \underset{(v',h'):\ \Delta F(v',h')>0}{\arg\max} \Delta F(v',h') \quad .$$

(v,h) is added to the set of accepted histories, and all m–grams (v,h,w) are added to the model. We then extend (v,h) by each of its seen predecessor words u in the training corpus and add the histories (u,v,h) to the set of candidate histories. Then the selection procedure is repeated. To make the selection procedure more robust and faster, only those (v,h) with $N(v,h)$ above a given count threshold N_T are considered as candidate histories.

The final varigram model uses the same parameter estimation scheme and the same discounting scheme as the trigram model, with the discounting values d_h pooled over all histories of the same length.

3. ADAPTIVE TOPIC–DEPENDENT MODEL

Usual static language models are trained on text corpora consisting of many different topics. On the test corpus, they do not adapt their probabilities according to the topic of the actual test data. Most often this problem is addressed by a cache language model which is made an additional component of interpolated language models, as in [4] and [8]. Additionally, in this paper we interpolate topic–dependent language models with the baseline model and perform an adaptation by putting more weight on that topic–dependent language model which is most promising given the current history, as in [7]. Further recent approaches to adaptive topic–dependent language modelling are [2] and [6].

3.1. Clustering Texts

The topic–specific trigram language models are trained on topic–specific partitions of the whole training corpus. To form these topic–specific partitions, we cluster the articles of the training corpus into C article clusters, one for each topic–specific partition, assuming that a single article represents a single topic. The clustering criterion is to maximize the log–likelihood F of the topic–specific unigram language models $q(w|c)$, $1 \le c \le C$, over all mapping functions $a \rightarrow c(a)$:

$$F = \sum_{a=1}^{A} \sum_{n=1}^{N_a} \log q(w_{a,n}|c(a))$$

with A as the number of articles in the training corpus, N_a as the length of article a, $w_{a,n}$ as the nth word of article a, and $c(a)$ as the article cluster of article a. We start the clustering process by randomly assigning one of the C article clusters to each article. Then we perform an exchange algorithm similar to [9], trying each article cluster for each article, and finally assigning the article to the best fitting article cluster. An alternative would be a bottom–up clustering algorithm as in [1]. However, in preliminary experiments, the exchange algorithm turned out to be more robust.

3.2. Parameter Adaptation

As described in [7], we use the adaptive linear interpolation of C topic–specific trigram language models together with a bigram cache model and a trigram model trained on the whole training corpus, resulting into $I = (C+2)$ interpolated language models:

$$p(w_n|h_n) = \sum_{i=1}^{I} \lambda_n(i) \cdot p_i(w_n|h_n)$$

with $\lambda_n(i)$ as the adaptive interpolation factor at position n for the ith language model $p_i(w_n|h_n)$. After the prediction of a word w_n at position n in the test corpus, we use w_n together with its $(M-1)$ predecessor words to reestimate the I adaptive interpolation parameters $\lambda_{n+1}(i)$ for position $(n+1)$ by using the well–known EM algorithm [3][7]. From preliminary experiments we noted that just one iteration of the EM algorithm achieves the lowest perplexities, resulting into the following update formula:

$$\lambda_{n+1}(i) = \frac{1}{M} \sum_{m=0}^{M-1} \frac{\lambda_n(i) \cdot p_i(w_{n-m}|h_{n-m})}{\sum_{j=1}^{I} \lambda_n(j) \cdot p_j(w_{n-m}|h_{n-m})} \quad .$$

M is set to the number of words from the start of the current article up to and including w_n, limited to the 500 most recent words. Strictly speaking, the notation should be M_n instead of M.

3.3. Binary Interpolated Models

With the adaptive model introduced above, all topic–specific models contribute to the prediction of word w_n at once. For a comparison, we tried a simpler model where only one of the topic dependent models is interpolated at one time with the baseline trigram and cache model. In lieu of one big interpolated model as in Section 3.2, this results in a set of adaptive binary interpolated models. The interpolation parameter $\mu_n(i)$ at corpus position n within the ith binary interpolated model is adapted as the $\lambda_n(i)$ described above. Let $\tilde{p}_i(w_n|h_n)$ be the ith binary interpolated model, $1 \le i \le C$, and $p_0(w_n|h_n)$ the baseline model. Then the ith binary interpolated model is given by the formula

$$\tilde{p}_i(w_n|h_n) = (1-\mu_n(i))p_i(w_n|h_n) + \mu_n(i)p_0(w_n|h_n) \quad .$$

Table 1: Test set perplexities of the trigram, L1O–selected varigram and cache–extended models.

	4M	39M
Word Trigram	152.1	96.8
+ Cache	120.5	82.8
Word Varigram	148.4	88.2
+ Cache	117.3	75.3

Table 2: Number of histories selected by the standard maximum likelihood vs. L1O selection scheme, resulting number of added varigrams, and test set perplexity for the varigram model ($N_T = 1$).

L1O	Histories		Varigrams		PP_{Test}	
	4M	39M	4M	39M	4M	39M
no	10 000	900 000	255 416	9 101 930	150.1	89.0
yes	10 000	300 000	80 892	2 148 328	148.4	88.2

The probability of the whole model is given by selecting the binary interpolated model with the lowest perplexity or, equivalently, the highest log–likelihood over the last M words:

$$\begin{aligned} p(w_n|h_n) &= \tilde{p}_{i_n}(w_n|h_n) \\ i_{n+1} &= \arg\max_i \sum_{m=0}^{M-1} \log \tilde{p}_i(w_{n-m}|h_{n-m}) \quad , \end{aligned}$$

where M is defined as in Section 3.2.

4. EXPERIMENTAL RESULTS

4.1. Perplexity Results

For the experiments, we used about 39 million words from the Wall Street Journal Corpus with the official 20K–word open non–verbalized punctuation vocabulary, resulting into about 14 million distinct trigrams. To examine the effect on a smaller corpus, we further used a subset of about 4 million words, resulting into about 2.3 million distinct trigrams. For the test, we used a corpus of about 325 000 words not included in the training corpora. The unknown word is included in the perplexity computation.

Compared to the word trigram model in Table 1, the varigram model has a large perplexity reduction on the 39M corpus and only a slight reduction on the 4M corpus. Thus, the selection schemes work on statistically highly significant data only. Adding the cache, the perplexity decreases further. The varigram and the cache models can be combined with almost no negative side effects in perplexity reduction: On the 39M corpus, the gain for the cache model is 14 %, for the varigram 9 %, and for the combined varigram and cache models 22 %, as compared to the trigram model. Compared to the combined trigram and cache models, the combined varigram and cache models achieve a reduction in perplexity of 9 %. Table 2 shows that the L1O selection scheme for varigram models achieves slightly better perplexity results as compared to the standard maximum likelihood selection scheme even at a drastically reduced number of selected histories.

Table 3: Test set perplexities of the adaptive model.

	C	4M	39M
Word Trigram	10	130.9	81.2
+ Adaptation	20	127.8	78.4
	30	126.7	76.7
Word Trigram	10	111.5	73.7
+ Adaptation	20	110.4	72.0
+ Cache	30	110.3	71.0
Word Varigram	10	109.3	69.5
+ Adaptation	20	107.8	67.6
+ Cache	30	107.6	66.7

Table 4: Test set perplexities of the binary interpolated and cache models.

C	4M	39M
10	112.3	75.5
20	111.2	74.0
30	111.2	72.9

For the adaptive models, we clustered our corpora into $C = 10, 20$, and 30 subsets of articles, respectively. Thus we made 10, 20, or 30 topic–specific models. We carried out experiments combining these topic–specific models with the trigram, varigram and cache models. The results are shown in Table 3.

Compared to the interpolated baseline trigram and cache model, we achieve a perplexity reduction of 9 % on the 39M corpus for the word–based varigram model with cache, of 14 % for the adaptive topic–dependent model using the baseline trigram and cache model, and of 19 % for the adaptive topic–dependent model using the varigram model with cache. The perplexities on the 4M corpus are also reduced, but not that much. Actually, using even smaller corpora results in almost no perplexity improvements. This clearly shows that both extensions are especially suited for large corpora.

Table 4 shows the results for the binary models. The topic–specific models are interpolated each with the baseline trigram and cache model. It is obvious that there must be a loss compared to the full adaptive models, but it is less than 3 %. So it should not be necessary to combine two or more of our topic–specific language models.

Table 5: Graph densities and graph errors of the word graph [11] (WGD – word graph density, NGD – node graph density, BGD – boundary graph density, GER – graph word error rate).

graph densities			graph errors [%]	
WGD	NGD	BGD	del/ins	GER
1476.21	181.80	19.67	0.1/0.5	4.2

Table 6: Effect of adaptation on the perplexities (PP) and recognition error rates (del/ins, WER) for word trigram and cache models ($C = 20$).

Adaptation	PP	rec. errors [%]	
		del/ins	WER
no	106.3	1.6/2.5	13.4
yes	97.6	1.8/2.4	13.6

4.2. Recognition Experiments

After having good perplexity results, we tried the adaptive model in the automatic speech recognizing system of RWTH [11]. Therefore, we used the 1994 ARPA recognition task on the North American Business Corpus (NAB). This corpus consists of about 240 million running words. We clustered it to generate 20 topic specific models for the adaptive topic dependent model. Due to the memory costs we used compact models, which treat all singleton trigrams as zero counts. With this language model, we ran a word graph rescoring. The word graph rescoring was carried out on the H1 development corpus including 310 sentences with 7387 words by 10 male and 10 female speakers. 199 of the spoken words were out–of–vocabulary words relative to the 20 000–word vocabulary. Some statistics of our word graph are summarized in Table 5 ([11], Table VIII, $F_{LAT} = 300$).

The recognition results reported in Table 6 are disappointing, in spite of the good perplexities. Our impression from an inspection of the NAB data is that the effect of topic adaptation is outweighted by other more important error sources. We will perform a detailed analysis of these errors in the next future.

5. CONCLUSIONS

In this paper, we have presented two extensions of the standard interpolated word trigram and cache model, namely the extension of the trigram model by useful word m–grams resulting into a varigram model, and the addition of topic–specific trigram models. We have presented the criteria for selecting useful m–grams and for partitioning the training corpus into topic–specific subcorpora. We have applied both extensions, separately and in combination, to corpora of 4 and 39 million words taken from the Wall Street Journal Corpus and achieved high reductions in perplexity of up to 19 % on the largest corpus. However, there have been almost no effects on the recognition results as compared to the standard trigram model with cache.

6. REFERENCES

[1] J. R. Bellegarda, J. W. Butzberger, Y.–L. Chow, N. B. Coccaro, D. Naik: *"A Novel Word Clustering Algorithm Based on Latent Semantik Analysis"*, Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, Atlanta, GA, Vol. I, pp. 172–175, May 1996.

[2] P. R. Clarkson, A. J. Robinson: *"Language Model Adaptation Using Mixtures and An Exponentially Decaying Cache"*, Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, Munich, Vol. II, pp. 799–802, April 1997.

[3] A. P. Dempster, N. M. Laird, D. B. Rubin: "Maximum Likelihood from Incomplete Data via the EM Algorithm", *J. Royal Statist. Soc. Ser. B (methodological)*, Vol. 39, pp. 1–38, 1977.

[4] M. Generet, H. Ney, F. Wessel: *"Extensions of Absolute Discounting for Language Modelling"*, Fourth European Conference on Speech Communication and Technology, Madrid, pp. 1245–1248, Sep. 1995.

[5] R. Kneser: *"Statistical Language Modeling Using a Variable Context Length"*, Fourth International Conference on Spoken Language Processing, Philadelphia, PA, Vol. 1, pp. 494–497, Oct. 1996.

[6] R. Kneser, J. Peters: *"Semantic Clustering for Adaptive Language Modeling"*, Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing, Munich, Vol. II, pp. 779–782, April 1997.

[7] R. Kneser, V. Steinbiss: *"On the Dynamic Adaptation of Stochastic Language Models"*, IEEE International Conference on Acoustics, Speech, and Signal Processing, Minneapolis, MN, Vol. 2, pp. 586–589, April 1993.

[8] R. Kuhn, R. de Mori: "A Cache–Based Natural Language Model for Speech Recognition", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 12, pp. 570–583, June 1990.

[9] S. C. Martin, J. Liermann, H. Ney: *"Algorithms for Bigram and Trigram Word Clustering"*, Fourth European Conference on Speech Communication and Technology, Madrid, pp. 1253–1256, Sep. 1995.

[10] T. R. Niesler, P. C. Woodland: *"A Variable–length Category–based N–gram Language Model"*, IEEE International Conference on Acoustics, Speech, and Signal Processing, Atlanta, GA, Vol. 1, pp. 164–167, May 1996.

[11] S. Ortmanns, H. Ney, X. Aubert: "A Word Graph Algorithm for Large Vocabulary Continuous Speech Recognition", *Computer, Speech and Language*, Vol. 11, No. 1, pp. 43–72, Jan. 1997.