Selection Criteria for Word Trigger Pairs in Language Modeling

Christoph Tillmann and Hermann Ney

Lehrstuhl für Informatik VI, RWTH Aachen – University of Technology
D-52056 Aachen, Germany
{tillmann,ney}@informatik.rwth-aachen.de

Abstract. In this paper, we study selection criteria for the use of word trigger pairs in statistical language modeling. A word trigger pair is defined as a long-distance word pair. To select the most significant trigger pairs, we need suitable criteria which are the topics of this paper. We extend a baseline language model by a single word trigger pair and use the perplexity of this extended language model as selection criterion. This extension is applied to all possible trigger pairs, the number of which is the square of the vocabulary size. When using a unigram language model as baseline model, this approach produces the mutual information criterion used in [7, 11]. The more interesting case is to use this criterion for a more powerful model such as a bigram/trigram model with a cache. We study different variants for including word trigger pairs into such a language model. This approach produced better word trigger pairs than the usual mutual information criterion. When used on the Wall Street Journal corpus, the trigger pairs selected reduced the perplexity of a full language model (trigram/cache) from 138 to 128 for a 5 million word training set and from 92 to 87 for a 38 million word training set.

1 Introduction

In speech recognition, the most widely used and successful language model is the so-called N-gram model, e. g. a bigram or trigram model, where the dependency of the word under consideration is limited to the immediate predecessor words. However it is clear that some sort of long-distance dependencies exist as well. The main goal in this paper is to include long-distance dependencies into the language model by means of so-called "trigger pairs" [7, 11]. In this work, we restrict ourselves to trigger pairs where both the triggered and the triggering events are single words (as opposed to word phrases). Unlike the approach presented in [1, 7], where the trigger pairs are selected on the basis of a mutual information criterion, the selection criterion presented in this paper is directly the perplexity improvement obtained by extending the baseline language model by a single trigger pair. What makes the selection criteria for word pair triggers interesting in general, is the following broader view: Given a baseline language model, how can we improve this model by including additional types of dependencies? For the selection criterion, we consider two variants. In the first variant, we directly combine trigger pairs with a given baseline model using a backing-off scheme

[6]. When using a unigram language model as baseline model, this approach produces the mutual information criterion used by Rosenfeld in [11].

The second variant we examined is based on the idea that trigger pairs in a language model are important to the extend they can improve a given baseline model. We thus adapted the selection criterion to exploit dependencies for trigger pairs beyond what is really supplied by a given baseline model. We proved that there are such dependencies.

Section 2 covers the mathematical models of the two selection criteria presented in this paper. In Section 3 we present the main experimental results. Examples of the trigger pairs, which were computed by the different methods, are presented. The identity of these examples significantly varies for the different methods. In the last section perplexity results are presented, where a trigram model with cache is improved by trigger pairs. The perplexity improvements achieved with the trigger pairs selected by the criteria presented in [7, 11] were much smaller.

2 Selection Criteria for Trigger Pairs

The goal of this paper is to reduce the perplexity of a given baseline language model $p(w|h)$ by means of word trigger pairs. $p(w|h)$ stands for a full language model, where the word w is predicted by the history h, which consists of the preceding words at a given position in the corpus. From the V^2 trigger pair candidates, where V is the number of words in the vocabulary, those trigger pairs are selected that best improve $p(w|h)$. The selection criteria are in terms of the direct perplexity improvement by a trigger pair on $p(w|h)$. This approach to select a trigger pair to extend a given model can be compared to the so-called feature selection in [2]. We present two new selection criteria: *high level trigger* and *low level trigger* selection.

2.1 High Level Triggers

In order to select a trigger pair, we fix a long distance trigger pair (a, b) and define an improved model $p_{ab}(w|h)$:

$$p_{ab}(w|h) = \begin{cases} q(b|a) & \text{if } a \in h \text{ and } w = b \\ [1 - q(b|a)] \cdot \dfrac{p(w|h)}{\sum\limits_{w' \neq b} p(w'|h)} & \text{if } a \in h \text{ and } w \neq b \\ q(b|\bar{a}) & \text{if } a \notin h \text{ and } w = b \\ [1 - q(b|\bar{a})] \cdot \dfrac{p(w|h)}{\sum\limits_{w' \neq b} p(w'|h)} & \text{if } a \notin h \text{ and } w \neq b \end{cases}, \qquad (1)$$

where $q(b|a)$ and $q(b|\bar{a})$ are two interaction parameters. Note, that for symmetry reasons we have introduced a special interaction parameter $q(b|\bar{a})$, when a has

not been seen in the history. The $q(b|a)$ and $q(b|\bar{a})$ are chosen to maximize the likelihood of the training corpus given the model $p_{ab}(w|h)$.

We now consider the difference between the log-perplexity F_{ab} of $p_{ab}(w|h)$ and the log-perplexity F_0 of the baseline model $p(w|h)$ on a corpus of size N. The aim is to find simpler expressions to calculate $F_{ab} - F_0$.

$$\begin{aligned} F_{ab} - F_0 &= \sum_{n=1}^{N} \log \frac{p_{ab}(w_n|h_n)}{p(w_n|h_n)} \\ &= \sum_{h} \Big[N(a;h,b) \log \frac{q(b|a)}{p(b|h)} \quad + \quad N(a;h,\bar{b}) \log \frac{1-q(b|a)}{1-p(b|h)} \\ &+ \; N(\bar{a};h,b) \log \frac{q(b|\bar{a})}{p(b|h)} \quad + \quad N(\bar{a};h,\bar{b}) \log \frac{1-q(b|\bar{a})}{1-p(b|h)} \quad \Big], \end{aligned} \tag{2}$$

where the count $N(a;h,b)$ is to be read as: number of occurences of the word b with history h in the training corpus so that h includes the word a. Multiple occurences of a are counted only once. Disregarding the dependency on h we define:

$$\begin{aligned} N(a;b) &= \sum_{n:a\in h_n, b=w_n} 1 \\ N(a;\bar{b}) &= \sum_{n:a\in h_n, b\neq w_n} 1 \\ N(\bar{a};b) &= \sum_{n:a\notin h_n, b=w_n} 1 \\ N(\bar{a};\bar{b}) &= \sum_{n:a\notin h_n, b\neq w_n} 1, \end{aligned} \tag{3}$$

where $a \in h_n$ means that the word a occured in history of word w_n. Because of the trigger pairs being combined with the baseline model in the above backing-off scheme on the highest level, we called the trigger pairs obtained by this method *high level triggers*.

For implementation purposes we found it convenient to rewrite $F_{ab} - F_0$ as follows. As a result of the backing-off scheme in Eq. (1) we can separate the effect of one trigger pair and a baseline model. We use the counts defined in defined by the Eq. (3) to rewrite $F_{ab} - F_0$ as follows:

$$\begin{aligned} F_{ab} - F_0 &= N(a;b) \log q(b|a) + N(a;\bar{b}) \log[1-q(b|a)] \\ &+ N(\bar{a};b) \log q(b|\bar{a}) + N(\bar{a};\bar{b}) \log[1-q(b|\bar{a})] \\ &- S(b) \end{aligned} \tag{4}$$

where $S(b)$ is:

$$S(b) = \sum_{n=1}^{N} \delta(w_n,b) \log \frac{p(b|h_n)}{1-p(b|h_n)} + \sum_{n=1}^{N} \log[1-p(b|h_n)]. \tag{5}$$

It is interesting to note that $S(b)$ is independent of the triggering word a. From this representation we draw the conclusions:

- We obtain maximum-likelihood estimates for the $q(b|a)$ by taking the derivates in Eq. (4) and setting the resulting equation to zero:

$$q(b|a) = \frac{N(a,b)}{N(a,b) + N(a,\overline{b})} \tag{6}$$

$$q(b|\bar{a}) = \frac{N(\overline{a},b)}{N(\overline{a},b) + N(\overline{a},\overline{b})}, \tag{7}$$

- If we fix a triggered word b and consider the triggering words a_i, $i = 1, 2, \cdots$ then the ranking of the a_i does not depend on the identity of the baseline model.

Implementation. The real problem of computing Eq. (4) is the second term in Eq. (5). It amounts to computing a corpus perplexity for each word in the vocabulary. To manage this computational problem, we used sampling. Typically we took every 20-th word to compute the sum $\sum_{n=1}^{N} \log[1 - p(b|h_n)]$. We compared this sampling approximation with the exact calculation on a training corpus of 5 million words and found for the tested words that sampling works quite well. To calculate the perplexity improvement $F_{ab} - F_0$ for the high level triggers, we first compute $S(b)$ for each triggered word b by using sampling. Secondly we need the trigger counts. An index structure is used, containing for each word a the positions of its occurrence in the corpus. For each triggering word a, we have to run once through all its positions in the corpus to get all the counts we need to compute the log-perplexity $F_{ab} - F_0$. For the following two criteria no sampling is needed, because ...

2.2 Unigram Triggers

Using a unigram model $p(w)$ as baseline model $p(w|h)$ we get:

$$\begin{aligned} F_{ab} - F_0 \;=\; & N(a;b) \log \frac{q(b|a)}{p(b)} \;+\; N(a;\bar{b}) \log \frac{1 - q(b|a)}{1 - p(b)} \\ & + \; N(\bar{a};b) \log \frac{1 - q(b|\bar{a})}{p(b)} \;+\; N(\bar{a};\bar{b}) \log \frac{1 - q(b|\bar{a})}{1 - p(b)} \end{aligned} \tag{8}$$

If we multiply Eq. (8) by $1/N$ and suppose $p(a;b) = \frac{N(a;b)}{N}$ we get exactly the mutual information criterion, used in [7, 11]. Thus this criterion is simply the improvement on the log-perplexity of a unigram model by the above backing-off model for one trigger pair. The trigger pairs selected by this criterion are called *unigram trigger*.

2.3 Low Level Triggers

Considering the model defined so far, there might be a drawback due to the following fact. The probability $q(b|a)$ in definition (1) is used whenever $w_n = b$ and $a \in h_n$, disregarding the probably high value of $p(w_n|h_n)$. The experimental results suggested another approach: To use trigger pairs only in positions where the probability $p_S(w_n|h_n)$ of a specific language model is less than a treshhold. We define an interpolated model as follows:

$$p(w|h) = [1-\lambda] \cdot p_S(w|h) + \lambda \cdot \beta(w)$$

where $\beta(w)$ is the unigram distribution of the words w in the corpus. We replace $\beta(w)$ by a new distribution $\beta_{ab}(w)$, incorporating a trigger pair $a \rightarrow b$ to produce a new model $p_{ab}(w|h)$. For the words w_n in positions n there is actually no difference between $p_{ab}(w_n|h_n)$ and $p(w_n|h_n)$, if $p_S(w_n|h_n)$ is greater than a threshold p_0. We define:

$$V(h) = \{w : p_F(w|h) > p_0\}$$

where p_0 is a probability threshold. For the difference $F_{ab} - F_0$ in the log-likelihoods, we obtain the approximation:

$$\begin{aligned} F_{ab} - F_0 &= \sum_{n=1}^{N} \log \frac{p_{ab}(w_n|h_n)}{p(w_n|h_n)} \\ &= \left[\sum_{n:\, w_n \notin V(h_n)} + \sum_{n:\, w_n \in V(h_n)} \right] \log \frac{p_{ab}(w_n|h_n)}{p(w_n|h_n)} \\ &\cong \sum_{n:\, w_n \notin V(h_n)} \log \frac{p_{ab}(w_n|h_n)}{p(w_n|h_n)} \\ &\cong \sum_{n:\, w_n \notin V(h_n)} \log \frac{\beta_{ab}(w_n|h_n)}{\beta(w_n)}, \end{aligned} \tag{9}$$

The low level triggers are selected, using $F_{ab} - F_0$. Using the approximation (9) this amounts in using a reduced corpus, cosnsisting of all positions n, where $w_n \notin V(h_n)$. β_{ab} is defined as in Eq. (1), where $p(w)$ is the unigram distribution of the reduced corpus. The trigger interaction parameters $q(b|a)$ and $q(b|\overline{a})$ are estimated on the reduced corpus, too. These trigger pairs we call *low level triggers* to oppose them to the high level triggers. The words $w_n \in V(h_n)$ are omitted as triggered events. But we allowed those words w_n to trigger words following in the corpus. This was done to have efficient data to get reliable trigger counts.

3 Experimental Results

We computed trigger pairs for three selection criteria:

A: unigram selection criterion in Eq. (8)
B: high level selection criterion in Eq. (1)
C: low level selection criterion in Eq. (9).

For the experiments we used training corpora from the Wall Street Journal task (WSJ task) [10]. There were three different corpora of 1, 5 and 38 million words. In the first part of this section we present samples of the selected pairs for the three criteria. They were computed on the 38 million word corpus. These samples we found typical after having gone through hundreds of examples of trigger pairs. In the second part we present perplexity results on test data.

3.1 Examples of Trigger Pairs

Considering trigger pairs, where triggering and triggered event are single words, we generally have V^2 candidates, where V is the size of the used vocabulary. Only trigger pairs that co-occured at least 3 times in a window of length 200 were used to calculated the perplexity improvement $F_{ab} - F_0$ according to the different criteria. For the unigram and low level triggers to carry out the calculation for all the above candidates took a maximum of 6 hours on the 38 million corpus to compute all trigger pair perplexities (on our Silicon Graphics Workstations with R 4000 processors). For the high level triggers the computation time was dominated by the need of sampling and depended on the sampling rate. We thus present for all three methods the best trigger pairs out of V^2 candidates.

As far as WSJ task mainly consists of financial texts and the trigger pairs from this domain dominate. Two tables show samples of trigger pairs obtained. Three lists of the best trigger pairs according to the three criteria are given in Table 1. For all three methods same-root triggers of the type $a \rightarrow a's$ and $a \rightarrow as$, where a noun a triggers its possesive $a's$ or its plural as, dominate. These trigger pairs have been removed, to single out the more interesting ones. Therefore the first column of Table 1 shows the position of the trigger pair within the original list. The second column presents the perplexity improvement of the extended model compared with the baseline model. The baseline model for the unigram triggers is a unigram model, for the low level and high level triggers it is a bigram model with cache. The four counts at the end of each line are the counts defined in Eq. (3).
Table 2 shows the best triggered words b for a number of triggering words a. The words b are ordered by decreasing perplexity improvement of the trigger pair $a \rightarrow b$. The trigger pairs are taken from lists of the best 500 000 for each method. We now discuss the two new selection criteria in greater detail:

High Level Triggers. We found the results for the high level triggers less satisfactory than for the low level triggers, but there are some interesting facts to note with high level triggers, too. There are some trigger pairs $a \rightarrow b$, where the bigram (b, a) is seen in the corpus, e.g. "Fe $\rightarrow$ Santa" The trigger pair $b \rightarrow a$ does not occur, because the corresponding word a is already predicted well by

Table 1. List of best word trigger pairs for the three selection criteria A, B and C (self triggers and same-root triggers excluded).

	Rank	$\delta PP3$	a	b	$N(a,b)$	$N(a,\overline{b})$	$N(\overline{a},b)$	$N(\overline{a},\overline{b})$
A	3	-2.22	the	a	839783	31263065	6175	3901023
	4	-2.21	a	share	15107	33430899	39833	2524207
	5	-1.75	in	nineteen	72010	33119066	54615	2764355
	11	-1.45	point	dollars	174009	14577007	66658	21192372
	12	-1.44	of	the	1793280	31921904	246783	2048079
	13	-1.41	the	company	75945	32026903	58876	3848322
	14	-1.29	the	U.	49630	32053218	47096	3860102
	16	-1.22	a	the	1985329	31460677	54734	2509306
	17	-1.17	the	of	767430	31335418	197787	3709411
	18	-1.10	percent	point	149707	12327082	92944	23440313
	19	-1.06	to	be	112112	33569246	44121	2284567
	20	-1.03	the	S.	80343	32022505	50732	3856466
	26	-0.96	the	company's	4693	32098155	19640	3887558
	27	-0.95	rose	point	65694	3275140	176957	32492255
	28	-0.95	in	the	1778846	31412230	261217	2557753
	29	-0.94	dollars	million	128117	16221255	42062	19618612
	32	-0.90	the	to	895618	31207230	36689	3870509
	33	-0.89	nine	point	122853	9678103	119798	26089292
	37	-0.86	dollars	cents	53792	16295580	4828	19655846
B	18	-0.0103	Texaco	Pennzoil	1423	294204	433	35713986
	19	-0.0102	Pennzoil	Texaco	1911	152412	2312	35853411
	30	-0.0074	Fe	Santa	1111	95276	1379	35912280
	34	-0.0071	distillers	Guinness	835	79004	802	35929405
	38	-0.0064	Am	Pan	1241	346056	975	35661774
	41	-0.0062	Campeau	Federated	844	134468	542	35874192
	45	-0.0061	Cola	Coca	807	144817	634	35863788
	64	-0.0051	oil	Opec	2274	2138246	221	33869305
	72	-0.0048	Federated	Campeau	941	129385	856	35878864
	107	-0.0039	multiples	negotiable	367	54612	86	35954981
	130	-0.0035	Geller	Lord	494	26838	652	35982062
	131	-0.0035	Beazer	Koppers	262	25132	131	35984521
	137	-0.0034	soviet	Moscow	1712	1173777	663	34833894
	163	-0.0031	rales	Interco	243	22795	147	35986861
	165	-0.0031	Eddie	crazy	478	67269	565	35941734
	171	-0.0030	Arabia	Saudi	802	147960	1145	35860139
	181	-0.0029	Warner	Borg	345	204029	132	35805540
	182	-0.0029	Shield	Robins	731	104266	517	35904532
	190	-0.0028	Robins	Dalkon	295	80880	40	35928831
	192	-0.0028	Shoreham	Lilco	247	29555	146	35980098
C	1	-0.00371	neither	nor	411	28775	567	1853529
	14	-0.00109	tip	iceberg	55	4944	4	1878279
	15	-0.00107	soviet	Moscow's	119	80652	26	1802485
	26	-0.00101	named	succeeds	147	63692	164	1819279
	27	-0.00100	Iraq	Baghdad	74	13766	45	1869397
	33	-0.00093	Eastman	Kodak's	49	3919	16	1879298
	40	-0.00090	Eastman	photographic	55	3913	61	1879253
	43	-0.00089	Carbide	Danbury	51	3350	46	1879835
	50	-0.00088	Eurodollar	syndication	60	3758	139	1879325
	55	-0.00086	filed	alleges	103	52441	80	1830658
	57	-0.00085	asked	replied	120	67419	110	1815633
	60	-0.00085	Kodak	photographic	57	6367	59	1876799
	68	-0.00083	motor	Ford's	74	25221	47	1857940
	71	-0.00083	South	Pretoria	87	71047	18	1812130
	75	-0.00080	Iran	Baghdad	80	42050	39	1841113
	76	-0.00080	occupational	Osha	40	3011	12	1880219
	80	-0.00079	soviet	Moscow	100	80671	45	1802466
	81	-0.00079	machines	Armonk	68	29004	28	1854182
	86	-0.00077	Peabody	Kidder's	49	8388	22	1874823

Table 2. List of best triggered words b for some triggering words a for the selection criteria A, B and C.

a	b
asked	A: point replied Mr. I he percent asked one seven eight B: Deltona Prism Benequity Taiyo Ropak Genesis Quintessential Envirodyne target's Teamster C: replied answered responded refused replies responses reply yes request requesting
airlines	A: airlines airline air passenger fares carriers traffic flights miles continental B: Delta's Northwest's Maxsaver Transtar Swissair Primark United's Motown Airbus's Cathay C: American's passengers Airlines' Eastern's United's hubs fares Northwest's carriers flights
buy	A: buy shares stock dollars company price offer million share stake B: Sheller Deltona Motown Northview Barren Philipp Selkirk Oshkosh Radnor Bumble C: repurchased Landover purchases repurchases Kohlberg repurchase Southland's undervalued
concerto	A: orchestra concerto music symphony piano violin philharmonic ballet composer concert B: Mozart violin Bach poignant C: strings orchestra violin score Mozart pianist recordings keyboard listen variations
Ford	A: Ford Ford's cars auto Chrysler car G. Jaguar models M. B: Ford Ford's Edsel ambulances Dearborn Jaguar Bronco Mustang Jaguar's Sheller C: Ford's Dearborn Bronco Taurus Escort Chrysler's Tempo Mustang Thunderbird subcompact
love	A: her love she point his I said dollars percent You B: Genex polly soothing boyish pathetic authenticity quaint Horace chalk Domino's C: beautifully passion sweet sexy romantic hero pop lovers pale wit
Microsoft	A: Microsoft software Lotus computer Microsoft's Apple computers personal O. one B: Microsoft Microsoft's Borland Ashton Lotus's Adobe Oracle Redmond Novell Bausch C: Microsoft's Redmond Apple's Borland spreadsheets Ashton Lotus's database spreadsheet
says	A: says said point million dollars adds seven he five one B: Benham Barren accredited Philipp Panasonic Radnor Deltona kids' Battelle Motown C: concedes explains adds agrees recalls asks insists acknowledges asserts predicts

the bigram model. In Table 1 the high level triggers only consist of proper names. Looking at the text all of them seem reasonable within the domain of Wall Street Journal business texts. Table 2 shows that the high level method fails to produce meaningful trigger pairs in some cases. An interesting fact to notice with high level triggers is that only 3000 out of V^2 possible trigger pairs were able to improve a given bigram model with cache. This is because the current word is predicted by a trigger pair with no regard to whether it is already predicted well by the bigram model with cache or not. From all this we draw the conclusion that trigger effects in general tend to be too weak to improve on a full baseline model in a backing-off fashion presented in this paper and that one should prefer a scheme, where a choice is made for when to use trigger pairs. This is done with the low level trigger pairs as introduced in this paper.

Low Level Triggers. In both tables the low level trigger pairs yield the best results in most cases. To understand them you sometimes have to take a close look at the underlying corpus, consisting of business texts. Some words produce very interesting trigger pairs, e.g. the verbs "asked" and "says" that mostly trigger verbs again, which even agree with them in tense. Another interesting example are the nouns "airlines" and "Ford", where the corresponding low level triggers show names of airlines or names of car models build by "Ford". The corresponding unigram triggers look worse for verbs, but for some nouns they

seem to have a kind of generalization capability in some cases.

The low level triggers resulted from using counts from a reduced corpus. It consisted of all positions of a given corpus of 38 million words for which a baseline model $p(w|h)$ computed a probability less than a given threshold $p_0 = 0.8 * 1/V$, where V is the number of words in the vocabulary. The baseline model was a bigram model trained on the same corpus , which was interpolated with a cache component with a weight of 0.1. Using that threshold 1.8 million positions were left, where the actual history h did not provide sufficient information with the baseline model $p(w|h)$ for the actual word w and where we want to rely on trigger pairs. We used different thresholds, but changing them has only a small effect on the selection of the calculated trigger pairs or the perplexity results. We emphasize the following facts with low level triggers:

- Among the best low level triggers are nouns that trigger their possessives, while self triggers do not occur at all.
- As well as using a probability threshold the corpus could be reduced by using only corpus positions n where the corresponding bigram (w_n, w_{n-1}) was seen only once and where w_n was not contained in the history h_n. The resulting pairs look very much the same.
- If we confine the history to the current sentence, we get trigger pairs, showing more grammatical structure, e.g. "I → myself", "We → ourselves". These results can be compared to the link grammar results in [4], where the grammar consists simply of pair of words.

The choice of pairs being used to extend a full language model depends on the model to be extended. The unigram trigger might offer a greater average usefulness in terms of mutual information, but the low level triggers have been selected to improve a full language model, consisting of bigram and cache. The perplexity results prove that they manage to provide information that supplements the information by bigram and cache.

3.2 Perplexity Results

In this subsection we present perplexity results which were achieved with the calculated trigger pairs on a trigram model with cache. We used the following model to incorporate the selected trigger pairs into a full language model:

$$p(w_n|h_n) = (1 - \lambda_1 - \lambda_2) \cdot p_S(w_n|h_n) + \lambda_1 \cdot p_C(w_n|h_n) + \lambda_2 \cdot p_T(w_n|h_n) \quad .$$

where the history $h_n = w_{n-M}^{n-1}$ consists of the M predeccesor words of w_n. The cache probability $p_C(w_n|w_{n-M}^{n-1})$ is defined as:

$$p_C(w_n|w_{n-M}^{n-1}) = \frac{1}{M} \sum_{m=1}^{M} \delta(w_n|w_{n-m}) \quad ,$$

with $\delta(w,v) = 1$ if and only if $w = v$. The trigger model is defined as:

$$p_T(w_n|w_{n-M}^{n-1}) = \frac{1}{M}\sum_{m=1}^{M}\alpha(w_n|w_{n-m}) \quad .$$

The $\alpha(b|a)$ are obtained by renormalization:

$$\alpha(b|a) = \frac{q(b|a)}{\sum\limits_{b'} q(b'|a)},$$

where the $q(b|a)$ are the maximum likelihood estimates as defined in Eq. (7). This renormalization is due to the fact that not all computed trigger pairs are used in a trigger model. In the experiments the history h consisted of all those words starting from the last article delimiter.

Perplexities were computed using a corpus of 325 000 words from the WSJ task. We used the computed word pairs together with a cache in an interpolated model. The λ_i in Eq. (10) were adjusted by trial and error in informal experiments. They can be trained by the EM procedure [3, 5]. The baseline trigram model was a backing-off model presented in [9]. We choose a number of the best trigger pairs as judged by the different selection criteria. We suppose that the combination of these trigger pairs will yield the best perplexity improvement within the model defined in Eq. (10). The problem with all the selection criteria presented is that the combination of the selected trigger pairs into one global language model is not captured by any of the criteria. However the low level criterion provides a better approximation to the use of the trigger pairs in Eq. (10). As opposed to the low level triggers, the high level triggers were not able to achieve perplexity improvements because the model defined in Eq. (10) is inadequate. In a first simple experiment we try to improve on a unigram model with

Table 3. Perplexity results for a unigram language model (5 million training words) with triggers and cache.

model	5 Mio
unigram	1027
+ low level triggers	960
+ unigram triggers	860
+ cache	750

the unigram triggers and the low level triggers in Table 3. The unigran model was trained on the 5 million corpus. We used the 500 000 best trigger pairs for low level and unigram triggers. The unigram triggers improve on an that unigram model to a much higher extend than the low level triggers can do. This is because the unigram triggers were selected to improve on an unigram model, whereas the low level triggers were selected to improve on a trigram model with

Table 4. Perplexity results for a trigram language model (1,5 and 38 million trainings words) with triggers and cache.

model	Number of Pairs	1 Mio	5 Mio	38 Mio
trigram with no cache		252	168	105
trigram/cache		197	138	92
+ unigram triggers	1500000	191	135	91
+ low level triggers	500000	182	130	88
+ low level triggers	1500000	180	128	87

cache. On the other hand the low level triggers were capable of improving on a trigram model with cache, which could not be achieved by using the original unigram triggers as shown in Table 4. The experiments with the trigram language model were carried out for different numbers of trigger pairs. The second column shows the number of the used trigger pairs. Using unigram triggers we weren't capable of achieving the same improvements as with the low level triggers.

The best results were obtained by employing the best 1.5 million trigger pairs. They prove that the low level triggers improve the trigram model with cache. Using 500 000 instead of 1 500 000 low level triggers only slightly changes the results.

4 Summary

In this paper, we considered the problem of selecting trigger pair pairs for language modeling. Rather than using some more or less arbitrary selection criterion, we presented a new method for finding word trigger pairs: given a reference language model to start with, we extend it by including a word trigger pair and compute the perplexity improvement of this extended model over the reference model. This perplexity improvement is used as selection criterion. For the special case of a unigram reference model, this new method is identical with the mutual information criterion. In the experimental tests, we found that the new method produces better results:

1. The selection criterion for the low level triggers produces intuitively better word trigger pairs than the usual mutual information criterion.
2. When used in a full language model, consisting of trigram model and cache the introduced low level triggers reduce the perplexity from 138 to 128 for the 5-million training set and from 92 to 87 for the 38-million training set. In comparison, when using the conventional mutual information criterion, the perplexity improvements were significantly smaller.

References

1. L.R. Bahl, F. Jelinek, R.L. Mercer and A. Nadas. "Next Word Statistical Predictor". *IBM Techn. Disclosure Bulletin*, 27(7A), pp. 3941–3942, 1984.

2. A. Berger, S. Della Pietra and V. Della Pietra. "A Maximum Entropy Approach to Natural Language Processing". In *Computational Linguistics*, Vol. 22, No. 1, pp. 39-71, March 1996.
3. A.P. Dempster, N.M. Laird and D.B. Rubin. "Maximum Likelihood from Incomplete Data via the EM Algorithm". In *Journal of the Royal Statistical Society*, Vol. 39, No. 1, pp. 1-38, 1977.
4. S. Della Pietra, V. Della Pietra, J. Gillett, J. Lafferty, H. Printz and L. Ures. "Inference and Estimation of a Long-Range Trigram Model". In *Lecture Notes in Artificial Intelligence*, Grammatical Inference and Applications, ICGI-94, Alicante, Spain, Springer-Verlag, pp. 78–92, September 1994.
5. F. Jelinek. "Self-Organized Language Modeling for Speech Recognition". In *Readings in Speech Recognition*, A. Waibel and K.F. Lee (eds.), pp. 450–506, Morgan–Kaufmann, 1991.
6. S.M. Katz. "Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer". In *IEEE Trans. on Acoustics, Speech and Signal Processing*, Vol. 35, pp. 400–401, March 1987.
7. R. Lau, R. Rosenfeld and S. Roukos. "Trigger-Based Language Models: A Maximum Entropy Approach". In *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Minnesota, MN, pp. II 45–48, April 1993.
8. R. Lau, R. Rosenfeld and S. Roukos. "Adaptive Language Modeling Using the Maximum Entropy Approach". In *Proceedings of the ARPA Human Language Technology Workshop*, pp. 108-113, Morgan–Kaufmann, March 1993.
9. H. Ney, M. Generet and F. Wessel. "Extensions of Absolute Discounting for Language Modeling". In *Fourth European Conference on Speech Communication and Technology*, pp. 1245–1248, Madrid, September 1995.
10. D.B. Paul and J.B. Baker. "The Design for the Wall Street Journal-based CSR Corpus". In *Proceedings of the DARPA SLS Workshop*, pp. 357–361, February 1992.
11. R. Rosenfeld. "Adaptive Statistical Language Modeling: A Maximum Entropy Approach". *Ph.D. thesis*, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, CMU–CS–94–138, 1994.

This article was processed using the LaTeX macro package with LLNCS style