INFORMATICA, 2006, Vol. 17, No. 1, 111–124

Cache-based Statistical Language Models of English and Highly Inflected Lithuanian

Airenas VAIČIŪNAS, Gailius RAŠKINIS
Department of Applied Informatics, Vytautas Magnus University
Vileikos 8, LT-44404 Kaunas, Lithuania
e-mail: airenas@freemail.lt, g.raskinis@if.vdu.lt

Received: August 2005

Abstract. This paper investigates a variety of statistical cache-based language models built upon three corpora: English, Lithuanian, and Lithuanian base forms. The impact of the cache size, type of the decay function, including custom corpus derived functions, and interpolation technique (static vs. dynamic) on the perplexity of a language model is studied. The best results are achieved by models consisting of 3 components: standard 3-gram, decaying cache 1-gram and decaying cache 2-gram that are joined together by means of linear interpolation using the technique of dynamic weight update. Such a model led up to 36% and 43% perplexity improvement with respect to the 3-gram baseline for Lithuanian words and Lithuanian word base forms respectively. The best language model of English led up to a 16% perplexity improvement. This suggests that cache-based modeling is of greater utility for the free word order highly inflected languages.

Key words: language models, n-grams, cache models, dynamic interpolation, perplexity reduction, inflected language, free word order language, Lithuanian.

1. Introduction

Statistical language models (LM) have become key components for large vocabulary continuous speech recognition (LVCSR) systems. These models provide prior probabilities that are used to rate hypothesized sentences and to disambiguate their acoustical similarities.

During the last few decades, much experimental work has been done in the field of statistical language modeling covering widespread world languages such as English, French, and German. The most popular modeling techniques developed for those languages are known as n-grams. Although n-grams have shown a good performance, they are far from optimal because of false word independency assumption.

Lithuanian language modeling has started since 2002. Lithuanian has free word order and is highly inflected, i.e., new words are easily formed by inflectional affixation. These properties of a language result in difficulties of statistical modeling known as huge vocabulary size, model sparseness, high perplexity, and a high out-of-vocabulary (OOV) word rate. The attempts to overcome the abovementioned difficulties of Lithuanian included word parsing into stems and endings (Vaičiūnas and Raškinis, 2003) as well as class-based modeling and modeling by morphological decomposition (Vaičiūnas and Raškinis, 2004). In this paper, we investigate an alternative cache-based modeling. To our knowledge, the cache-based modeling of highly inflected free word order languages has not been attempted. The cache-based models presented in this paper are interesting in two respects. They are able to adapt dynamically to the text under investigation and they have the potential of catching dependencies spanning longer word sequences than n-grams do. The impact of the model architecture, cache size, type of the decay function, including custom corpus derived functions, and interpolation technique (static vs. dynamic) on the perplexity of a language model is studied. Cache language models of Lithuanian are compared to the corresponding English ones.

2. Related Work

Cache-based n-gram model for linguistic applications was first introduced by Kuhn and De Mori, (1988, 1990). It can be thought of as a usual n-gram Markov model trained on a relatively short history of recent words of some particular word w_i.

Let w_i be the $i-$th word of a text and let $h = w_{i-K}, \ldots, w_{i-1}$ denote the cache or history of w_i, where K is the size of the cache.

Let $C(h) \leqslant K$ be the number of words within h belonging to the chosen vocabulary V.

Let $C(w_i, h)$ be the number of occurrences of a word w_i within h.

Let $C(w_{i-1}, w_i, h)$ be the number of word pairs w_{i-1}, w_i within h.

Finally, let $I(condition)$ denote the indicator function taking the value 1 if *condition* is true and 0 otherwise.

Then, the conditional probabilities of 1-gram and 2-gram cache language models can be estimated by formulas (1) and (2) respectively:

$$\widehat{P}_H(w_i|h) = \frac{C(w_i, h)}{C(h)} = \frac{\sum_{j=i-K}^{i-1} I(w_i = w_j)}{\sum_{j=i-K}^{i-1} I(w_j \in V)}, \tag{1}$$

$$\widehat{P}_{H^2}(w_i|w_{i-1}, h) = \frac{C(w_{i-1}w_i, h)}{C(w_{i-1}, h)} = \frac{\sum_{j=i-K}^{i-2} I(w_{i-1} = w_j \wedge w_i = w_{j+1})}{\sum_{j=i-K}^{i-2} I(w_{i-1} = w_j)}. \tag{2}$$

Conditional probabilities of a 3-gram cache LM can be estimated in a similar way. Jelinek *et al.* (1991) showed that 2-gram and 3-gram cache outperformed 1-gram cache in terms of LM perplexity[1]. Rosenfeld (1996) and Goodman (2001) reported just minor improvements of 3-gram cache over the 2-gram cache. Nevertheless, 1-gram cache language models are often used because of the problem of LM sparseness arising due to the limited cache size K.

Clarkson and Robinson (1997) suggested an improvement to (1) and (2) based on the experimental evidence that the probability of a word reoccurrence in a text exponentially

[1] Perplexity refers to how many different equally probable words a statistical LM expects to appear in average for a particular type of a context. It is estimated on the test subset of the corpora.

decays as the distance to that word increases. Otherwise stated, recent words w_j have greater influence on probability distribution of the current word w_i. The influence diminishes as the distance $i-j$ increases. The decay cache is used to model this phenomenon:

$$\widehat{P}_{d(H)}(w_i|h) = \frac{\sum_{j=i-K}^{i-1}[I(w_i = w_j) \cdot d(i-j)]}{\sum_{j=i-K}^{i-1} d(i-j)}, \tag{3}$$

$$\widehat{P}_{d(H^2)}(w_i|w_{i-1,}h) = \frac{\sum_{j=i-K}^{i-2}[I(w_{i-1} = w_j \wedge w_i = w_{j+1}) \cdot d(i-j)]}{\sum_{j=i-K}^{i-2}[I(w_{i-1} = w_j) \cdot d(i-j)]}, \tag{4}$$

where $d(x)$ is the decay function that tends to zero as the distance x increases. Two exponentially decaying functions are often used: $d(x) = \mathrm{e}^{-bx}$ and $d(x) = a\mathrm{e}^{-bx}+c$. The decay speed b as well as parameters a and c are chosen experimentally or estimated by approximating the function of word reoccurrence, i.e., the actual histogram of distances between the two consecutive repetitions of the same word.

Because of a very limited cache size standalone cache models (1)–(4) are sparse and should be used in combination with n-gram models built upon larger text corpora. Linear interpolation is the most popular method of such combination (Jelinek *et al.*, 1991; Iyer and Ostendorf, 1999; Clarkson, 1999; Tillmann and Ney, 1996). 1-gram and 2-gram cache models can be linearly interpolated with the standard word 3-gram model $\widehat{P}_{W^3}(w_i|w_{i-2}, w_{i-1})$ in a way shown below:

$$\begin{aligned}\widehat{P}_{W^3+H}(w_i|w_{i-2}, w_{i-1}) &= \lambda \cdot \widehat{P}_{W^3}(w_i|w_{i-2}, w_{i-1}) \\ &\quad + (1-\lambda) \cdot \widehat{P}_H(w_i|h), \quad 0 \leqslant \lambda \leqslant 1, \end{aligned} \tag{5}$$

$$\begin{aligned}\widehat{P}_{W^3+H+H^2}(w_i|w_{i-2}, w_{i-1}) &= \lambda_{W^3}\widehat{P}_{W^3}(w_i|w_{i-2}, w_{i-1}) \\ &\quad + \lambda_H\widehat{P}_H(w_i|h) + \lambda_{H^2}\widehat{P}_{H^2}(w_i|w_{i-1}, h), \end{aligned} \tag{6}$$

$$0 \leqslant \lambda_{W^3}, \lambda_H, \lambda_{H^2} \leqslant 1, \quad \lambda_{W^3} + \lambda_H + \lambda_{H^2} = 1,$$

Here, λ's are interpolation weights optimized on the validation corpus.

Sometimes conditional interpolation formula is used (Goodman, 2001):

$$\widehat{P}_{W^3+H+[H^2]}(w_i|w_{i-2}, w_{i-1}) = \begin{cases} \widehat{P}_{W^3+H+H^2}(w_i|w_{i-2}, w_{i-1}), & \text{if } w_{i-1} \in h, \\ \widehat{P}_{W^3+H}(w_i|w_{i-2}, w_{i-1}), & \text{otherwise.} \end{cases} \tag{7}$$

Besides the standard word 3-gram models, cache models can be interpolated with class-based, skip, sentence mixture models (Goodman, 2001), topic mixture models (Kneser and Steinbiss, 1993; Iyer and Ostendorf, 1999), and trigger pair models (Tillmann and Ney, 1996).

Models (5)–(7) are based on the static interpolation weights λ_M. Dynamic model interpolation weights $\lambda_M(i, h_D)$ can also be used (Kneser and Steinbiss, 1993). The basic idea of this approach is that if the cache-hit[2] is small in recent history the weight of the general 3-gram component should be increased and vice versa.

[2] The percentage of words w_i of the test corpus such that $C(w_i, h) > 0$.

Dynamic weights may be adapted on a word by word basis by optimizing perplexity on the recent word history $h_D = w_{i-D}, \dots, w_{i-1}$:

$$\widehat{P}_{W^3 \oplus H \oplus H^2}(w_i|w_{i-2}, w_{i-1}) = \lambda_{W^3}(i, h_D)\widehat{P}_{W^3}(w_i|w_{i-2}, w_{i-1}) + \lambda_H(i, h_D)\widehat{P}_H(w_i|h) + \lambda_{H^2}(i, h_D)\widehat{P}_{H^2}(w_i|w_{i-1}, h), \tag{8}$$

where $\lambda_{W^3}(i, h_D) + \lambda_H(i, h_D) + \lambda_{H^2}(i, h_D) = 1$ for all i, and D represents the length of an empirically chosen word history. The $\lambda_M(i, h_D)$ can be estimated by an expectation maximization algorithm (see (Kneser and Steinbiss, 1993; Martin *et al.*, 1997) or (Gotoh and Renals, 1997)) before estimating the combined probability estimate (8). Dynamic interpolation was previously introduced in topic mixture models of highly inflected Slovenian (Maucec and Kacic, 2001) and Finish (Siivola *et al.*, 2001). Some other attempts to avoid using static interpolation weights include the definition of interpolation weights as the function of a cache size K (Goodman, 2001) and the use of distinct weights $\lambda(i)$ for classes of topic-specific and general purpose words (Federico and Bertoldi, 2001; Gotoh and Renals, 1997; Martin *et al.*, 1997; Seymore *et al.*, 1998).

There is no consensus about the efficiency of the cache-based LM embedded in a speech recognition system. Jelinek *et al.* (1991), Rosenfeld (1996), Tillmann and Ney (1996) reported WER[3] reduction, while Clarkson (1999) and Goodman (2001) reported WER degradation due to the use of a cache-based LM. In all those cases, the perplexity of the cache-based LM was significantly better than the perplexity of a word 3-gram LM.

3. Resources and Tools

Our investigations were based on three corpora. The main corpus was the Lithuanian text corpus compiled by the Center of Computational Linguistics at Vytautas Magnus University (Marcinkevičienė, 2000) containing 84 202 576 word tokens (henceforth LT corpus). This corpus represented a great variety of genres and topics of the present day written Lithuanian. It was used for the investigation of cache based language modeling phenomena of inflected Lithuanian. Two auxiliary corpora were the corpus of Lithuanian base forms (LTBF) and “The Sunday Times” English corpus of the year 1995 (EN). The LTBF corpus was derived from the LT corpus by replacing each word with its base form[4]. Auxiliary corpora served for inflected/non-inflected and Lithuanian/English comparison purposes.

All corpora were divided into training, validation and test subsets constituting 98%, 1%, and 1% of the original corpora respectively. The same proportions of text genres were kept within all subsets. We used some text clearing: punctuation was removed, numbers and out-of-vocabulary (OOV) words (i.e., words found in the test subset but absent from the vocabulary V) where replaced by tags $\langle num\rangle$ and $\langle oov\rangle$, respectively.

[3] Word Error Rate is the standard measure of accuracy of a speech recognition system.

[4] Base forms (the infinitive for verbs, the singular nominative case for nouns, etc.) were obtained with the morphological lemmatizer of Lithuanian (Zinkevičius, 2000). In case of morphological ambiguity, the first base form out of the list of possible base forms was selected.

Table 1
Summary of corpora characteristics

Corpus	Word types (vocabulary)	Word tokens			Articles	Average words per article
		training	validation	testing		
LT	1158k	84 202 k	853 k	713 k	1996	42185
LTBF	371k					
EN	235k	40 525 k	409 k	400 k	91167	445

Majority of our investigations were carried out using locally developed cache-based language modeling tools. Simple n-grams were built using CMU-Cambridge Statistical Language Modeling Toolkit (Clarkson and Rosenfeld, 1997) that was extended to handle vocabularies of more than 65k words.

4. Experimental Results

We have investigated cache-based models in order of increasing complexity. First, the simple 1-gram cache, 1-gram decaying cache, and 1-gram decaying cache using dynamic weight adaptation were investigated. Thereafter, the best performing 1-gram cache models were complemented with the components of 2-gram cache, 2-gram decaying cache, and 2-gram decaying cache using dynamic weight adaptation. Throughout all experiments, cache-based LMs were compared on the basis of perplexity and perplexity improvement with respect to the baseline. The results are briefly summarized in the Table 2. More detailed description of our investigations is presented in the subsections that follow.

Table 2
Summary of cache-based language modeling experiments

Language model	LT, 1157k	LTBF, 371k	EN, 235k
		Perplexity	
$\widehat{P}_{W^3}$ (3-gram baseline, Kneser-Ney)	1027.21	451.27	259.46
		Perplexity improvement, %	
$\widehat{P}_{W^3+H}$ (+1-gram cache)	24.72	30.64	12.24
$\widehat{P}_{W^3+d(H)}$ (+1-gram decaying cache)	28.20	34.24	13.49
$\widehat{P}_{W^3\oplus d(H)}$ (+ dynamic weight adaptation)	29.62	35.94	13.71
$\widehat{P}_{W^3+d(H)+[H^2]}$ (+2-gram cache, static weights)	33.51	39.55	15.69
$\widehat{P}_{W^3+d(H)+[d(H^2)]}$ (+2-gram decaying cache)	33.32	40.13	16.02
$\widehat{P}_{W^3\oplus d(H)\oplus[d(H^2)]}$ (+ dynamic weight adaptation)	36.21	43.03	16.20

Table 3

Perplexities and OOV rates of 3-gram language models obtained with Kneser-Ney and Good-Turing smoothing techniques

Corpus	Vocabulary size	Perplexity of $\widehat{P}_{W^3}$		OOV, %
		Kneser-Ney smoothing	Good-Turing smoothing	
LT	1157k	1027.21	1117.42	1.73
LTBF	371k	451.27	478.68	1.15
EN	235k	259.46	276.76	0.31

All experiments were carried out without cache flushing[5], as we wanted to investigate the ability of LMs to adapt to the changes in text topics. OOV handling was realized in the following way: terms of type $\widehat{P}(\langle oov\rangle|h)$ and $\widehat{P}(\langle oov\rangle|w_i, h)$ were skipped, but $\widehat{P}(w_i|\langle oov\rangle, h)$, were included into perplexity calculations.

4.1. *Choice of the Baseline Language Model*

We have chosen the conventional word-based 3-gram $\widehat{P}_{W^3}(w_i|w_{i-2}, w_{i-1})$ including all singleton 3-grams and smoothed using Kneser-Ney (Kneser and Ney, 1995) smoothing technique as our baseline model. Kneser-Ney smoothing systematically outperformed Katz backoff technique (Jelinek, 2001) coupled with Good-Turing smoothing (see Table 3).

4.2. *1-gram Cache-based Models*

We have constructed a series of 1-gram cache-based models $\widehat{P}_{W^3+H}$ for cache sizes K ranging from 50 to 1000. Perplexity improvement and cache hit for each K was measured. The obtained results are summarized by Fig. 1 and Fig. 2.

1-gram cache-based model significantly improved perplexity with respect to baseline $\widehat{P}_{W^3}$ by 12% (EN), 25% (LT) and 31% (LTBF). Perplexity improvement showed similar cache size dependency curves for both languages. The best improvements were achieved at $K = 300$ (EN and LTBF) and $K = 500$ (LT). Cache hit estimates confirmed our intuition that Lithuanian words were less used by the cache because of a bigger inflected vocabulary. Cache hit curve for LTBF was similar to that of EN, but the perplexity improvement for EN was much lower.

4.3. *1-gram Decaying Cache-based Models*

We have investigated 1-gram cache-based models $\widehat{P}_{W^3+d(H)}$ with four types of decay functions.

[5] The term “cache flushing” means that all words are removed from the cache at the end of an article.

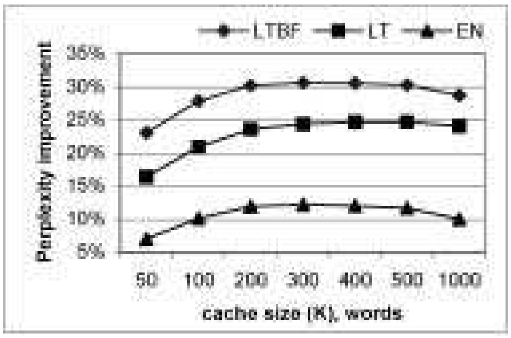

Fig. 1. Impact of the 1-gram cache size on the perplexity improvement.

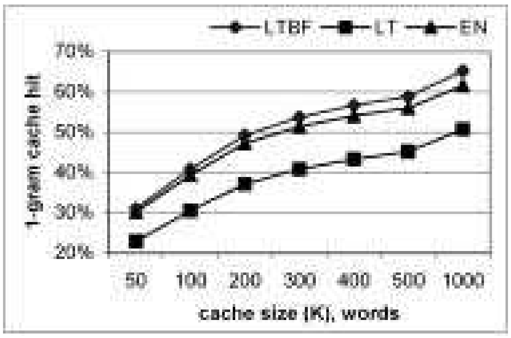

Fig. 2. Impact of the 1-gram cache size on the cache hit.

Exponential decay function $d_b^{\exp}(x) = \mathrm{e}^{-bx}.$ (9)

Linear decay function[6] $d_a^{linear}(x) = \max(a - x, 0).$ (10)

Gamma-like decay function[7] $d_{a;b}^{gamma}(x) = x^{a-1}\mathrm{e}^{-bx}.$ (11)

Corpus-derived decay function $d_o^{corpus}(x) = \sum_{i=x+1}^{N} I(w_i = w_{i-x} \wedge Occ(i,x) = o).$ (12)

Here, N is the size of the *corpus* and $Occ(i,x) = \sum_{j=i-x+1}^{i-1} I(w_i = w_j)$. The expression $(w_i = w_{i-x} \wedge Occ(i,x) = o)$ is true if and only if $w_i = w_{i-x}$ and there is exactly o occurrences of the same word in between w_i and w_{i-x}. Thus, the functions $d_0^{corpus}(x)$ and $d_1^{corpus}(x)$ represent respectively the histograms of distances between two consecutive and two next to consecutive repetitions of the same word[8].

[6] The linear decay function was included for comparison purposes only.

[7] The popular exponential decay function has maximum at the position one. But this contradicts empirical data as identical words rarely follow one another. Empirical evidence suggests that the probability of the word to reoccur grows from the start and then starts decaying after some position.

[8] In all decaying cache experiments, we used a discrete array implementation for storing function values. The maximum cache size K was truncated at $K = 1000$ for speed-up purposes. The values of $d(K)$ are relative small for $K > 1000$.

Optimum parameters for the decay functions (9, 11) were found experimentally, by optimizing perplexity on the validation data set. Corpus-derived decay functions were individually estimated on training subsets of LT, LTBF and EN corpora. Adding decay to 1-gram cache resulted in an improvement of about 3.5% for Lithuanian and 1.3% for English models.

The optimum cache size and decay speed were inversely related. Thus, slower decaying functions were used for the LT task as it had longer caches. It is interesting to note that $d_{0.01}^{\exp}(x)$ was one of the best decay functions for EN corpus and actually had a decay speed different from the decay speed of the distribution of word reoccurrences $d_0^{EN}(x)$ (Fig. 3a, 3d). Nevertheless, taking into account the second reoccurrence of the word could be of some help for both languages (Table 4, last line). It is also interesting to note that

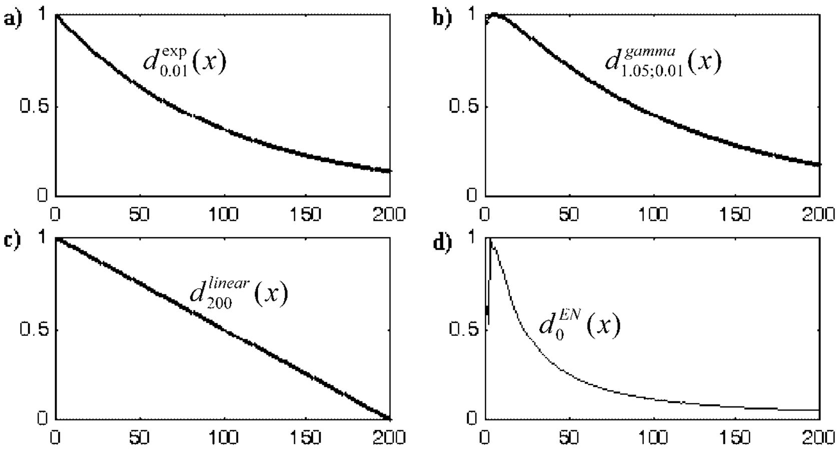

Fig. 3. Sample decay functions normalized by dividing by the maximum.

Table 4

Perplexity improvements obtained with various decay function types of 1-gram cache

Decay function, $d(x)$	Perplexity of $\widehat{P}_{W^3+d(H)}$		
	LT (1157k)	LTBF (371k)	EN (235k)
None (best cache size)	773.31 (500)	313.00 (300)	227.71 (300)
$d_{500}^{linear}(x)$	757.52	305.20	225.90
$d_{0.015}^{\exp}(x)$	756.83	302.84	225.79
$d_{0.01}^{\exp}(x)$	746.70	299.23	224.60
$d_{0.005}^{\exp}(x)$	742.13	299.15	224.99
$d_{0.0025}^{\exp}(x)$	750.90	305.06	227.40
$d_{1.05;0.005}^{gamma}(x)$	743.22	299.88	225.19
$d_{1.10;0.01}^{gamma}(x)$	746.23	299.25	224.47
$d_0^{corpus}(x)$	737.56	**296.07**	225.17
$d_0^{corpus}(x) + d_1^{corpus}(x)$	**737.49**	296.14	**224.46**

the distribution of word reoccurrences $d_0^{EN}(x)$ and $d_0^{LT}(x)$ of English and Lithuanian appeared to be very similar. This distribution seems to be a language independent parameter.

4.4. *2-gram Cache-based Models*

We have constructed a series of 2-gram cache-based models $\widehat{P}_{W^3+d(H)+[H^2]}(7)$ for cache sizes K ranging from 50 to 50000. 2-gram cache-based model $\widehat{P}_{W^3+d(H)+[H^2]}$ significantly outperformed 1-gram model $\widehat{P}_{W^3+d(H)}$ (having $d(x) = d_0^{corpus}(x) + d_1^{corpus}(x)$) by 5% (LT, LTBF), and 2% (EN). The optimum K value was about 30000, 2000 and 500 words for LT, LTBF and EN corpora respectively. Important differences in optimum K values could be explained by the fact that the average article size is more than 40k words in LT and only 445 words in EN corpus.

Adding decay to the cache 2-gram improved LTBF and EN models, but not the LT model (Table 5). This can be explained by the fact that decay functions used "truncated" cache size of $K = 1000$, much less than the optimum cache size for LT models. Corpus derived decay functions analogous to (12) seem to be best suited for Lithuanian corpora and exponential decay works best for the English corpus.

An interesting fact is that 2-gram cache hit on LTBF and even on LT was larger than on EN texts (see Fig. 4). This can probably explain why 2-gram cache improves English LMs not as much as Lithuanian LMs.

4.5. *Dynamic Adaptation of Component Weights*

We have constructed a series of 1-gram and 2-gram cache-based models of type $\widehat{P}_{W^3 \oplus d(H)}$ and $\widehat{P}_{W^3 \oplus d(H) \oplus [d(H^2)]}(8)$ for D ranging from 20 to 500. As it was expected, dynamic weight adaptation outperformed static weight optimization. The model

Table 5

Perplexity improvements obtained with various decay function types of 2-gram cache

Decay function, $d(x)$	Perplexity of $\widehat{P}_{W^3+d(H)+[d_2(H^2)]}$		
	LT (1157k)	LTBF (371k)	EN (235k)
None (best cache size)	**683.04** (30000)	272.81 (2000)	218.76 (500)
$d_{1000}^{linear}(x)$	687.06	271.32	218.22
$d_{0.015}^{\exp}(x)$	688.71	272.30	218.23
$d_{0.01}^{\exp}(x)$	687.01	271.29	217.88
$d_{0.005}^{\exp}(x)$	685.48	270.51	**217.67**
$d_{0.0025}^{\exp}(x)$	686.10	270.79	217.99
$d_{1.05;0.005}^{gamma}(x)$	685.55	270.54	217.69
$d_{1.10;0.01}^{gamma}(x)$	686.85	271.29	217.83
$d_0^{corpus}(x)$	685.11	270.28	217.76
$d_0^{corpus}(x) + d_1^{corpus}(x)$	**684.97**	**270.19**	217.89

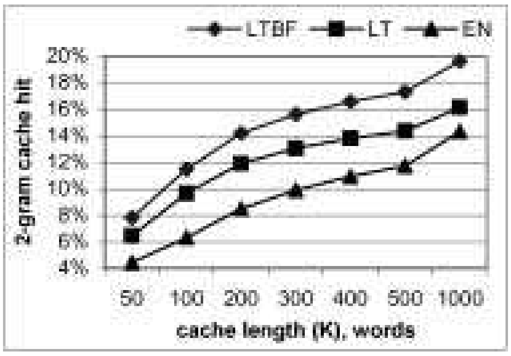

Fig. 4. Impact of the 2-gram cache size on the cache hit.

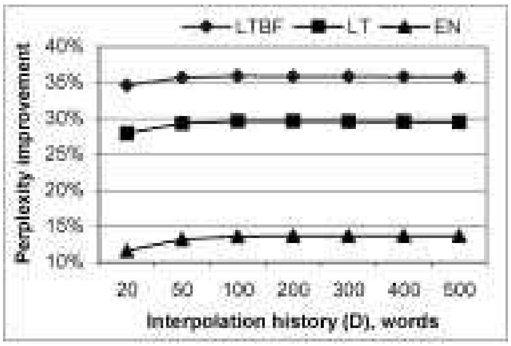

Fig. 5. Impact of the size of interpolation optimization history D on the perplexity of $\widehat{P}_{W^3 \oplus d(H)}$.

$\widehat{P}_{W^3 \oplus d(H) \oplus [d(H^2)]}$ added about 3% (LT, LTBF) and 0.2% (EN) of improvement. Tiny improvement of LMs built over EN corpus was probably due to the shortness of EN articles. The 2-gram cache component of EN models had its utility as well as its average weight reduced. Thus, weight adaptation procedure could bring little gain over static weights. In contrary, 2-gram cache component was extremely useful for some articles of LT and LTBF corpora. In this case, the dynamic weight adaptation boosted LM performance.

Though short interpolation optimization histories h_D had the potential of better adaptation to the changes in article or text topic, there were no perplexity improvements for short histories ($D < 50$ words) because of small reliability of such short histories. The optimum history size was found to be $D = 200$ for both 1-gram and 2-gram models for all three corpora. The perplexity grew slowly for $D > 200$ (see Fig. 5).

It is interesting to note that LMs using dynamic weight adaptation had different average component weights for texts belonging to different stylistic categories (Table 6). For instance, legal documents had average $\lambda_{H^2}(i, h_D) = 0.2$ indicating repeated usage of word-pair collocations.

Table 6

Average weights of $\widehat{P}_{W^3 \oplus d(H) \oplus d(H^2)}$ components per text category of LT corpus

Text category	Average weights of $\widehat{P}_{W^3 \oplus d(H) \oplus [d(H^2)]}(8)$ components		
	$\lambda_{W^3}(i, h_D)$	$\lambda_H(i, h_D)$	$\lambda_{H^2}(i, h_D)$
National newspapers	0.88	0.11	0.01
Translated philosophy	0.70	0.23	0.07
Legal documents	0.75	0.05	0.20

4.6. *Other Approaches Related Cache-based Modeling*

Rosenfeld (1996) found that the reduction of the perplexity could be achieved by using the cache for rare words only. Such cache usage appeared not to be useful to Lithuanian. However, we found that some perplexity reduction could be gained by omitting certain “unpromising” words of the validation corpus from the cache (boosting the probabilities of remaining words). The unpromising words were defined as those having average probability estimate given by $\widehat{P}_{W^3+H}$ lower than $\widehat{P}_{W^3}$, i.e., words w having *perf*(w) less than some negative constant, where

$$perf(w) = \sum_{i=1}^{N_{Val}} \big(\log_2 \widehat{P}_{W^3+H}(w_i|w_{i-2}, w_{i-1}) \\ - \log_2 \widehat{P}_{W^3}(w_i|w_{i-2}, w_{i-1}) \big) \cdot I(w_i = w)$$

and the sum is over validation corpus. This approach resulted in some though negligible improvement.

5. Conclusions

In this paper, we described a number of experiments with the cache-based LMs of Lithuanian and English. Our work confirmed that significant reduction of perplexity (43.03%, 36.21% and 16.20% for LTBF, LT, and EN corpora respectively) could be achieved by the use of the cache-based modeling. Improvements over the baseline are higher than twice for Lithuanian with respect to English. English 3-gram baseline model performs relatively well and is hard to improve as English has a strict word order. Simplistic claim that “worse models can be better improved” cannot explain this difference. Actually, we repeated the whole set of experiments by replacing Kneser-Ney smoothed 3-grams with worse Good-Turing smoothed 3-grams. Perplexity improvement obtained with those worse models was the same as with the better ones through the whole set of experiments. This suggests that cache-based modeling brings more benefits to the free word order languages by being capable of capturing some dependencies that lie besides the strict word order.

Cache improved LMs of Lithuanian word base forms (LTBF) better than LMs of Lithuanian words (LT). This suggests that additional efficiency can be brought into language modeling of Lithuanian by methods that are able to cope with the highly inflected nature of Lithuanian.

The impact of different modeling techniques had similar tendencies in case of both languages. Adding 1-gram cache component to the 3-gram model brought the greatest part of improvement. Additional improvement was gained by adding a 2-gram cache component, by selecting an appropriate decay function, and by replacing static component interpolation weights with the procedure of dynamic weight update.

It was found that optimal decaying function differs from the distribution of the distances of the word reoccurrence, in general. However it is possible to construct better decay functions by analyzing longer relations, for example distribution of the distance to the second reoccurrence. It appeared that the best 1-gram cache size is independent of language, i.e., it is the same for EN and LTBF tasks. Experiments confirmed that longer cache size should be used in the 2-gram cache case. These last findings should be regarded with care because of the differences in article size in Lithuanian and English corpora.

This research confirms that cache-based modeling significantly improves LM perplexity. Our next task is to integrate them into a speech recognition system ant to investigate their impact on a speech recognition accuracy.

References

Clarkson, P. (1999). *Adaptation of Statistical Language Models for Automatic Speech Recognition*. PhD thesis, Cambridge University Engineering Department, Cambridge.

Clarkson, P., and R. Rosenfeld (1997). Statistical language modeling using the CMU-Cambridge Toolkit. In *Proceedings of 5th European Conference on Speech Communication and Technology*. pp. 2707–2710.

Clarkson, P., and A. Robinson (1997). Language model adaptation using mixtures and an exponentially decaying cache. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*. pp. 799–802.

Federico, M., and N. Bertoldi (2001). Broadcast news LM adaptation using contemporary texts. In *Proceedings of 7th European Conference on Speech Communication and Technology*, vol. A42. pp. 239–242.

Gildea, D., and T. Hofmann (1999). Topic-based language models using EM. In *Proceedings of 6th European Conference on Speech Communication and Technology*. pp. 2167–2170.

Goodman, J.T. (2001). A bit of progress in language modeling. *Computer Speech and Language*, **15**(4), 403–434.

Gotoh, Y., and S. Renals (1997). Document space models using latent semantic analysis. In *Proceedings of 5th European Conference on Speech Communication and Technology*. pp. 1443–1446.

Iyer, R., and M. Ostendorf (1999). Modeling long distance dependence in language: topic mixture vs. dynamic cache models. In *Proceedings of the IEEE Transactions on Speech and Audio Processing IEEE-SAP*, vol. 7. pp. 30–39.

Jelinek, F. (2001). *Statistical Methods for Speech Recognition*. Massachusetts Institute of Technology, Cambridge.

Jelinek, F., B. Merialdo, S. Roukos and M. Strauss (1991). A dynamic LM for speech recognition. In *Proceedings of the ARPA Workshop on Speech and Natural Language*. pp. 293–295.

Kneser, R., and H. Ney (1995). Improved backing-off for m-gram language modeling. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*. pp. 181–184.

Kneser, R., and V. Steinbiss (1993). On the dynamic adaptation of stochastic language models. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*. pp. 586–589.

Kuhn, R. (1988). Speech recognition and the frequency of recently used words: a modified Markov model for natural language. In *Proceedings of 12th International Conference on Computational Linguistics*. pp. 348–350.

Kuhn, R., and R. De Mori (1990). A cache-based natural language model for speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **12**(6), 570–583.

Marcinkevičienė, R. (2000). Corpus linguistics in theory and practice. *Darbai ir Dienos*, **24**, 7–64 (in Lithuanian).
`http://donelaitis.vdu.lt/`

Martin, S.C., J. Liermann and H. Ney (1997). Adaptive topic-dependent language modelling using word-based varigrams. In *Proceedings of 5th European Conference on Speech Communication and Technology*. pp. 1447–1450.

Rosenfeld, R. (1996). A maximum entropy approach to adaptive statistical language modeling. *Computer Speech and Language*, **10**, 187–228.

Sepesy Maucec, M., and Z. Kacic (2001). Topic detection for language model adaptation of highly-inflected languages by using a fuzzy comparison function. In *Proceedings of 7th European Conference on Speech Communication and Technology*, vol. A42. pp. 243–247.

Seymore, K., S. Chen and R. Rosenfeld (1998). Nonlinear interpolation of topic models for language model adaptation. In *Proceedings of ICSLP*-98.

Siivola, V., M. Kurimo and K. Lagus (2001). Large vocabulary statistical language modeling for continuous speech recognition in Finnish. In *Proceedings of 7th European Conference on Speech Communication and Technology*, vol. B25. pp. 737–741.

The Lithuanian Language Corpus at Vytautas Magnus University.
`http://donelaitis.vdu.lt/`

Tillmann, Ch., and H. Ney (1996). Statistical language modeling and word triggers. In *Proceedings of the SPECOM 96*. pp. 22–27.

Vaičiūnas, A., G. Raškinis and V. Kaminskas (2004). Statistical Language Models of Lithuanian based on word clustering and morphological decomposition. *Informatica*, **15**(4), 565–580.

Vaičiūnas, A., and G. Raškinis (2003). Statistical modeling of Lithuanian language. In *Proceedings of the Conference "Information Technologies 2003"*, KTU, Kaunas. pp. IX 35–40 (in Lithuanian).

Zinkevičius, V. (2000). Lemuoklis – tool for morphological analysis. *Darbai ir dienos*, 245–274 (in Lithuanian).

A. Vaičiūnas (born in 1976) received his MSc degree in computer science from the Vytautas Magnus University in Kaunas in 2000. Presently, he is a PhD student at the same university. His research interests are natural language modeling and speech recognition.

G. Raškinis (born in 1972) received his MSc degree in artificial intelligence and pattern recognition from the University of Pierre et Marie Curie in Paris in 1995. He received doctor's degree in the field of informatics (physical sciences) in 2000. Presently, he works at the Center of Computational Linguistics and teaches at the Department of Applied Informatics of VMU. His research interests include application of machine learning techniques to human language processing.

Statistiniai kalbos modeliai, naudojantys trumpalaikę atmintį, anglų ir lietuvių kalboms

Airenas VAIČIŪNAS, Gailius RAŠKINIS

Šiame straipsnyje aprašomi statistinių kalbos modelių, naudojančių trumpalaikę atmintį, tyrimai. Modeliai įvertinami naudojant tris skirtingus tekstynus: anglišką, lietuvišką ir lietuvišką pagrindinių formų tekstyną. Darbe pateikiama šių modelių maišaties priklausomybė nuo atminties žodžių kiekio, slopinančios funkcijos tipo, bei modelių interpoliavimo būdo (statinis arba dinaminis). Geriausi rezultatai buvo gauti naudojant dinamiškai interpoliuotus standartinį 3-gramos, netolimos praeities 1-gramos ir 2-gramos (su slopinančiomis funkcijomis) modelius. Naudojant trumpalaikės atminties modelius maišatis sumažėjo (lyginant su standartine 3-grama) 36% lietuviškam, 43% lietuviškam pagrindinių formų bei 16% angliškam tekstynui.