

# Methods for Combining Language Models in Speech Recognition

*Simo Broman and Mikko Kurimo*

Neural Networks Research Centre  
Helsinki University of Technology, Finland

Simo.Broman@hut.fi, Mikko.Kurimo@hut.fi

## Abstract

Statistical language models have a vital part in contemporary speech recognition systems and a lot of language models have been presented in the literature. The best results have been achieved when different language models have been used together. Several combination methods have been presented, but few comparisons of the different methods has been done.

In this work, three combination methods that have been used with language models are studied. In addition, a new approach based on likelihood density function estimation using histograms is presented. The methods are evaluated in speech recognition experiments and perplexity calculations. The test data consist of Finnish news articles and four language models work as the component models.

In the perplexity experiments, all combining methods produced statistically significant improvement compared to the 4-gram model that worked as a baseline. The best result, 46 % improvement to the 4-gram model, was achieved when combining three language models together by using the new bin estimation method. In the speech recognition experiments, 4 % reduction to the word error and over 7 % reduction to the phoneme error was achieved by unigram rescaling method.

## 1. Introduction

Language modeling has a vital part in contemporary speech recognition systems and many other areas of language technology including character recognition, machine translation, and spelling correction. In speech recognition, the task of the language model is to determine a probability for a word given the word history.

The most popular language modeling paradigm is the family of n-gram models. Though simple, n-gram models have proven to be powerful and hard to outperform. Lot of work has been done in developing models that would better utilize syntactic or semantic structure of the language. In literature, language models that model different aspects of language have successfully been combined together. A lot of combining methods have been presented but a thorough investigation of different combining methods has not been done.

The purpose of this work is to study different methods that have been used in combining language models. Also a new approach based on likelihood density function estimation is presented for combining language models.

In this work, four combining methods, linear interpolation, log-linear interpolation, unigram rescaling, and the new bin estimation method, are used in combining four language models. Some of the experiments are reproductions of works presented in the literature. However, the Finnish language and the use of morpheme-like sub-word units as the basic units bring in a fresh aspect.

## 2. Component language models

N-gram models are the most important language models and standard components in speech recognition systems. In this work, a Kneser-Ney smoothed 4-gram model was used as a reference and a component in all combinations.

A word that has occurred in the past is much more likely to re-occur in the near future than would be expected from its global frequency. A cache model [1] utilizes this phenomenon by keeping track of the words that have occurred and raising their probability estimates in the future. In this work, a bigram cache was used. Both unigrams and bigrams were stored into the cache and an ordinary bigram model was constructed from them.

The other large context language models, latent semantic analysis (LSA) [2] and topic model [3], try to capture long-range dependencies of words by presenting topic information as latent variables. In LSA the mathematically relevant part is the singular value decomposition that provides a vector presentation for words and documents. The distance between a word and a document is transformed to a probability estimate that depicts the semantic relation of the two. In the topic model, the word probability is calculated as

$$P(w|h) = \sum_{t=1}^T P(w|t)P(t|h), \quad (1)$$

where  $t$  is a latent variable that is supposed to refer to different topics.  $P(w|t)$  is the topic-specific probability distribution and  $P(t|h)$  is the mixing weight that depicts the semantic relation of the word history  $h$  and the word  $w$ . The topics are learned from the data using an EM-algorithm.

## 3. Morph based language models

The large number of words caused by inflection produces severe problems in language modeling. Any vocabulary of practical size cannot adequately cover all words, and the out-of-vocabulary-rate will be intolerably high. Also, the large vocabulary worsens the problem of data sparsity. To overcome these problems, we use an unsupervised word segmentation algorithm [4] to automatically split words into smaller units “morphs” that approximate the natural morphemes. Using the morphs, the size of the vocabulary can be reduced dramatically. The morphs are used as basic units in all language models in this work.

A problem that arises in speech recognition when using sub-word units instead of whole words is that we no longer know where the previous word ends and the next one begins. In most cases, continuous speech does not give any acoustical information about the word boundaries. So the word boundaries have to be determined by the language model. In the training

data the word boundaries are marked with a special symbol and treated as normal morphs.

## 4. Combination methods

### 4.1. Linear interpolation

A simple and widely used combining method is linear interpolation which simply means taking a weighted sum of the probabilities given by the component language models. The interpolation weights are optimized on the held-out data. An advantage of the linear interpolation is that it is simple and fast to calculate. If the inputs are probability estimates, also the output is a probability estimate.

### 4.2. Log-linear interpolation

Another simple combining method is log-linear interpolation, defined by equation

$$p(w_i) \sim \prod_{k=1}^K P_k(w_i)^{\lambda_k} \quad (2)$$

where  $\lambda_k$ 's are scaling parameters that adjust the contribution of each component model. The motivation behind the log-linear interpolation lies in the better synergy of the components compared to the linear interpolation. When both components are large, the log-linear interpolation yields a value larger than either of its components. Respectively, when both components are small, the result is smaller than the components. A drawback in the log-linear interpolation, as in all non-linear combination methods, is that the output is no longer a probability estimate as the sum of the values over all words is not necessarily one. Thus, the result should be normalized over all words in the vocabulary. However, the effect of the normalization in speech recognition is small and it is usually omitted.

### 4.3. Unigram rescaling

The unigram rescaling method is expressed by formula

$$p(w_i|h) \sim \frac{P_1(w_i|h)P_2(w_i|h)}{P(w_i)} \quad (3)$$

where  $P_1$  and  $P_2$  are the component model probabilities and  $P(w_i)$  is the normal unigram probability. In this work, exponential scaling terms were used to adjust the components and the unigram probability term. The method was presented in [3] for integrating the topic model with the n-gram model and it was shown to outperform the linear and the loglinear interpolation. In [2] the method was used in combining the LSA model with the n-gram model and it was shown that the Eq. 3 can be derived for the LSA model and the n-gram model under relatively mild assumptions.

### 4.4. Bin estimation method

The idea in the bin estimation method is to determine the likelihood for a word given the estimates from the component models. When combining two models which produce probability estimates  $P_A$  and  $P_B$  this can be written as

$$p(w = w_i|P_A(w = w_i|h), P_B(w = w_i|h)) \quad (4)$$

To estimate this from data, the input space spanned by the component model outputs is divided to rectangular bins in which the

likelihood is assumed to be constant. The likelihood inside bin  $B_r$  is estimated by estimator

$$\hat{p}_{B_r} = \frac{\sum_{k=1}^L I((P_A(w = w_k|h_k), P_B(w = w_k|h_k)) \in B_r)}{\sum_{j=1}^N \sum_{k=1}^L I((P_A(w = w_j|h_k), P_B(w = w_j|h_k)) \in B_r)} \quad (5)$$

where  $I(x)$  is index function whose value is 1 if  $x$  is true and 0 otherwise.  $w_k$  in the numerator refers to the  $k$ 'th word in the training data,  $h_k$  refers to the word sequence  $w_1 \dots w_{k-1}$ , and the outer summation in the denominator is calculated over the vocabulary. The derivation for the estimator is presented in [5]. In the test phase, the component model values are calculated and the corresponding bin is chosen to determine the likelihood which can be transformed to a probability estimate by normalizing over all words.

Using a constant likelihood value inside a bin introduces some quantization error. Tightening the grid would decrease the error but at the same time it would make the likelihood estimates in each bin more inaccurate as less data is left for each bin. Finding the optimal division means minimizing the error produced by these two error sources. In this work, 53x53 grid was used for all two-model combinations. Gentle filtering in two dimensions was applied for smoothing the values and for bins that had no data points a value was set by interpolating the neighboring bins. When using the bin-method with the topic model, the topic model values were divided by the unigram probability as in unigram rescaling method.

A drawback of the method is that it needs quite a large amount of data to properly estimate the likelihood function, especially with several models and a dense grid.

## 5. Experiments and results

All long-span models, i.e. topic, LSA, and cache model, are combined with the 4-gram model one at a time. The cache model was only used with the linear interpolation and the bin estimation method. In addition to the two model combinations, also some three and four model combinations are evaluated. Many of the three and four model combinations could be discarded based on the results of the two model combinations and only the most appealing combinations are evaluated.

The bin estimation method is easily generalized to the case of more than two models. However, the number of bins grows rapidly when more models are brought in. Here the bin estimation method was used to combine the 4-gram model, the topic model, and the cache model together. In this case a different formulation of the cache model, called here three-value-cache, was used to keep the number of the bins tolerable. In the three-value-cache only three values are used: 0: the word not in the cache, 1: the unigram is in the cache, and 2: the bigram is in the cache. The combination has still three times more bins than the two model combinations. To compensate this, three times more held-out data was used to train the combining parameters in this particular case. So this experiment was not strictly comparable to the other experiments. However, the other studied combining methods might not benefit from increasing the amount of training data since they have only few parameters that will probably achieve the optimal values or close enough with less data.

### 5.1. Perplexity

A text corpus of Finnish news articles from STT (Finnish National News Agency) was used for training and evaluating the language models and the combining methods. The text corpus

comprises about 16.4 million words in 91 000 articles. The text corpus was divided to three parts: training, development, and test set, consisting of 14 million, 165 000, and 200 000 words. The rest was left out for future purposes. The training set was used for training the language models, the development set for training the combination methods, and the test set for evaluating the combinations.

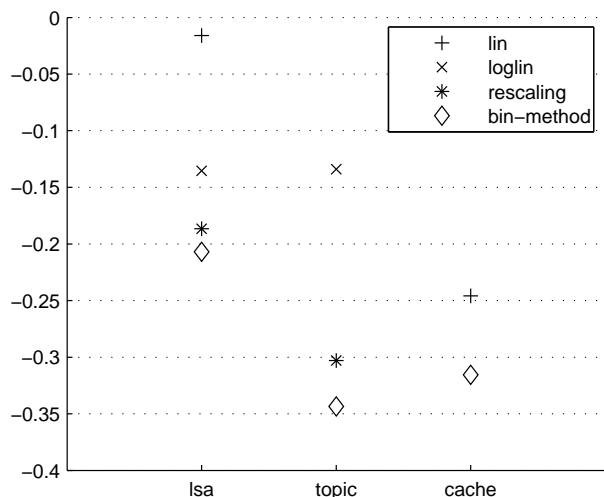


Figure 1: *Relative perplexities for two model combinations compared to the 4-gram baseline 5584. E.g. -0.35 means that perplexity has decreased by 35 %.*

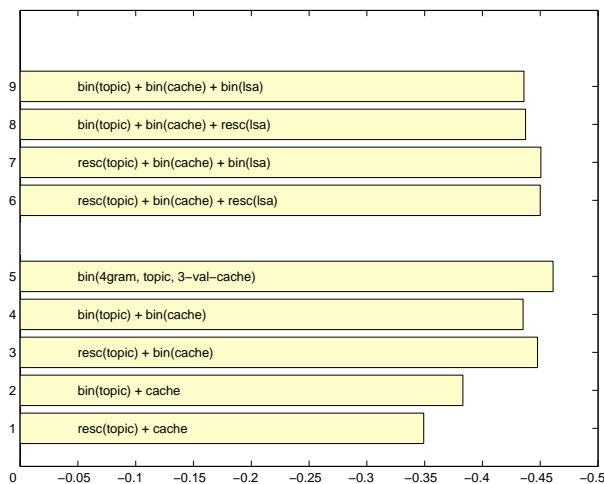


Figure 2: *Relative perplexities for three and four model combinations compared to the 4-gram baseline. 4-gram model is the other component in all pairwise combinations and is left out from the notation. So resc(topic) refers to the unigram rescaling combination of the topic and the 4-gram model. The plus (+) symbol denotes linear interpolation.*

The perplexity result for the plain 4-gram model was 5584. The perplexity is normalised by the number of whole words, instead of morphs, to make the perplexity independent of the chosen morph set. Because Finnish words are long and may consist of several morphemes, the perplexity is much higher than, for

example, for English. The relative perplexities compared to this baseline are presented in Figure 1.

The linear interpolation did not improve the result with the topic model but otherwise all combining methods yielded significant perplexity reduction over the plain 4-gram model. For all two-model combinations, the best performing method was the bin-method. The second best method, when applicable, was the unigram rescaling performing slightly worse than the bin-method. Notably worse was the log-linear interpolation achieving still 13 % improvement over the baseline. In all cases, the linear interpolation was the worst method producing only slight improvement over the baseline with the LSA model and no improvement with the topic model. However, with the cache model also the linear interpolation performed well producing nearly 25 % improvement. All these improvements to the baseline are statistically significant by the Wilcoxon signed-rank test.

The results concerning the topic and the LSA model are quite similar to the results reported by [3] and [6]. An exception is the linear interpolation that failed to produce any improvement when combining the topic model and the 4-gram model. This is probably due to the word boundaries that were also predicted by the models. The relative perplexity reductions for the three and four model combinations over the plain 4-gram model are depicted in Figure 2. In these experiments the two-model combinations have been further combined together using linear interpolation. An exception is the combination 5 in which 4-gram, topic, and cache model have been combined together using the bin estimation method.

The greatest perplexity reduction, 46 %, was achieved by combination 5 where the 4-gram model, topic model, and the cache model were all combined together using the bin estimation method. Almost equally good result, 44.8 %, was achieved by combination 3, in which the unigram rescaling combination of the 4-gram and the topic model was interpolated with the bin-method combination of the 4-gram and the cache model. Adding the LSA model to the combination of the 4-gram model, topic model, and the cache model yielded only little further improvement to the result. The LSA and the topic model focus in modeling the same aspect of the language. Thus, only little cumulative improvement is achieved when combining these two together.

## 5.2. Speech recognition

Speech recognition tests were run with the same model combinations as the perplexity experiments. The speech recognition was performed by HUT's continuous speech recognition system that applies unlimited vocabulary language modeling based on unsupervised morpheme-like subword units. We refer to [7] for the more detailed system descriptions and remind here only some features relevant to the current application. The baseline speech recognition system consists of MFCC feature extraction, HMM acoustic model, morph-based lexical model, morph n-gram language model, and a start-synchronous stack decoder. An advantage in the stack decoder is that it suites well for long-range language models. The recognized word history is maintained in a tree structure, so that each hypothesis has a uniquely defined word history. This allows us to easily integrate the examined long-range language models into the system. The state of the language model, i.e. the representation of the recognized word history, is stored together with the hypothesis. When the hypothesis is expanded with a new word, the state is updated and stored with the new hypothesis.

Speech recognition experiments were run with read STT news articles from years 1988-1992. The speech data consist of 288 articles of about one minute length. 3.7 hours was used in training the acoustic models, 30 minutes was used in tuning the combining method parameters and the language model scaling factor that adjusts the balance between the language model and the acoustic model, and 40 minutes was used for the tests.

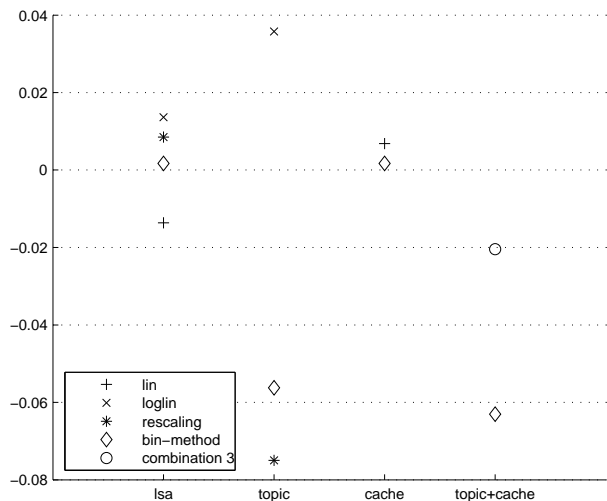


Figure 3: Relative phoneme error rate reductions compared to the 4-gram baseline 5.87 %.

Using the plain 4-gram model yielded 25.7 % word error rate and 5.87 % phoneme error rate in the speech recognition experiments. The relative phoneme error rates compared to this baseline are depicted in Figure 3. We see that the large perplexity reductions turn into rather small improvements in speech recognition results. While the bin-method was dominating in the perplexity experiments, such tendency is not observed in the speech recognition results. One reason for this is that the bin-method was not adjusted based on the speech data while the other methods were. So the results are not strictly comparable together. The best performing method varies depending on the model and also whether looking at the word or phoneme error rate. The significantly best performing combination was the unigram rescaling combination of the 4-gram and the topic model which achieved 4 % word error reduction and 7.5 % phoneme error reduction compared to the plain 4-gram model. This difference is also the only statistically significant improvement by the Wilcoxon test.

## 6. Conclusions and discussion

In this work, four combining methods: linear interpolation, log-linear interpolation, unigram rescaling, and bin estimation method were evaluated by perplexity and speech recognition experiments. In all experiments, the Kneser-Ney-smoothed 4-gram model was used as the baseline. In the perplexity experiments, the best performing method was the new bin estimation. The greatest overall perplexity reduction, 46 %, was achieved by using the bin estimation method in combining 4-gram, topic, and the cache models together. The result is one of the greatest perplexity reductions that has been reported over the properly smoothed 4-gram model. However, the remarkable perplexity reductions turned only into small improvements in the speech recognition experiments.

The presented bin estimation method is one possible implementation for using multivariate function estimation methods in the task of combining language models. It is left for future work to study whether improvements could be achieved by more thorough parameter optimization or using some other function estimation methods. In the bin estimation method, no assumptions are made about the models to be combined. Also, the only restriction set to the likelihood function is that it has to be slowly varying enough to be accurately estimated by the histograms. These things and the better performance compared to the other evaluated methods suggest that the bin-method may be applicable in combining many kind of models. However, to make further conclusions would need more experiments with different kind of language models. The disadvantage of the bin method is that it needs a large amount of data to train the combining parameters properly. This may restrict the number of the models that can be combined simultaneously with the method.

Probably the most commonly used combining method, the linear interpolation, is fast to calculate and the parameter estimation is easy as it has very little parameters and the heavy calculation of the normalization is avoided. In cases where the linear interpolation does not introduce severe averaging, it seems to have relatively good performance and it may be a good choice for the combining method. Some interesting methods, like maximum entropy approach, were left out for future work.

## 7. Acknowledgements

We thank Ms. Inger Ekman and the Department of Information Studies at the University of Tampere for the speech data. The work was supported by the Academy of Finland in the projects *New information processing principles* and *New adaptive and learning methods in speech recognition*. This work was also supported in part by PUMS/TEKES project and by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778.

## 8. References

- [1] R. Kuhn and R. de Mori, "A cache-based natural language model for speech recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 6, June 1990.
- [2] J. R. Bellegarda, "Exploiting latent semantic information in statistical language modeling," in *Proceedings of the IEEE*, vol. 88. IEEE, August 2000, pp. 1279–1296.
- [3] D. Gildea and T. Hofmann, "Topic-based language modeling using em," in *Proc. Eurospeech*, 1999, pp. 2167–2170.
- [4] M. Creutz and K. Lagus, "Unsupervised discovery of morphemes," in *Proceedings of the Workshop on Morphological and Phonological Learning of ACL-02*, Philadelphia, Pennsylvania, July 2002, pp. 21–30.
- [5] S. Broman, "Combining methods for language models in speech recognition," Master's thesis, Helsinki University of Technology, 2005.
- [6] N. Coccaro and D. Jurafsky, "Towards better integration of semantic predictors in statistical language modeling," in *ICSLP*. Sydney, Australia: ICSLP, November 1998.
- [7] T. Hirsimäki and M. Kurimo, "Decoder issues in unlimited finnish speech recognition," in *Proceedings of the 6th Nordic Signal Processing Symposium*, Espoo, Finland, June 9-11 2004, pp. 320–323.