LOG-LINEAR INTERPOLATION OF LANGUAGE MODELS

Dietrich Klakow
Philips GmbH Forschungslaboratorien,
Weißhausstr.2, D-52066 Aachen, Germany, klakow@pfa.research.philips.com

Abstract

A new method to combine language models is derived. This method of log-linear interpolation (LLI) is used for adaptation and for combining models of different context length. In both cases LLI is better than linear interpolation.

1 Introduction

Combining models of different nature is an important task in language modeling. Up to now, linear interpolation [1] has been the preferred method. It was used to combine models trained on different corpora or to add a cache-component to a M-gram model. Adaptive linear interpolation [2] is used for the same task but adjusts the interpolation weights during recognition.

A far more powerful method of combination are maximum-entropy models [3, 4]. There, individual features which should be included into the model are selected and used as constraints. During training, a model is constructed from the constraints and the final model satisfies all constraints simultaneously. However, this method suffers from very long training times and hence is not widely spread in application. In addition, the constraints have to be consistent [8], which means that an arbitrary selection of features may cause training to fail.

Thus, we suggest a new method called log-linear interpolation (LLI) which is related to maximum entropy models but has all the flexibility and the same number of free parameters as linear interpolation. This paper aims at a strict comparison of linear and log-linear interpolation under identical conditions the only difference being the method of combination. As an aside we also want to shed a light on various other issues discussed above.

2 Log-Linear Interpolation

This first section will show the formal equivalence of a constrained minimum Kulback-Leibler distance approach and a linear interpolation of scores. A set of N models p_i is given. Those models may be of any type (i.e. zerogram, unigram, ...) and also trained on different corpora. The unknown model will be denoted by p. It is now assumed that the Kullback–Leibler distance of the unknown model to the given models p_i is

$$D(p||p_i) = d_i \quad , \quad i = 1 \ldots N \quad . \tag{1}$$

An additional model p_0 is given and the distance to this model should be kept small. At the same time the constraints (1) should be satisfied. Using Lagrange multipliers λ'_i yields

$$\mathcal{D} = D(p||p_0) + \sum_i \lambda'_i (D(p||p_i) - d_i) \tag{2}$$

which has to be minimized with respect to p [6]. A closed solution of the model is easily obtained. To simplify the results, the Lagrange multipliers are redefined and now denoted by λ. In addition, the model is finally expressed in terms of conditional probabilities. To complete this derivation, p_0 is assumed to be a uniform distribution which allows p_0 to be absorbed into the normalization $Z_\lambda(h)$ of the model. Hence,

$$p(w|h) = \frac{1}{Z_\lambda(h)} \prod_i p_i(w|h)^{\lambda_i} \tag{3}$$

is obtained. Because of the special selection of p_0 and the redefiniton of λ_i no explicit constraint on the interpolation weights λ_i appears. The final model (3) can be interpreted as a linear interpolation of scores. However, an additional score from the normalization $Z_\lambda(h)$ has to be added.

This model will be compared to the well established linear interpolation [1]

$$p(w|h) = \sum_i p_i(w|h)\alpha_i \tag{4}$$

with $\sum_i \alpha_i = 1$. To ensure that $p(w|h)$ are probabilities, the α_i also have to be positive.

3 Training

Corresponding to the two different formal settings, there are two different schemes to calculate the free parameters d_i or λ_i. We decided to optimize the log-likelihood

$$F(\{\lambda_i\}) = \sum_{h\,w} f(h\,w) \log \left(\frac{1}{Z_\lambda(h)} \prod_i p_i(w|h)^{\lambda_i} \right) \tag{5}$$

and to optimize with respect to the λ_i. Here $f(h\,w)$ are the frequencies of the M-gram $h\,w$ in the cross-validation set.

First, it is important to observe that $F(\{\lambda_i\})$ is convex in all λ_i. To prove this, one may use the statements that the sum of convex functions is again a convex function and that applying a convex and monotonically increasing function to a convex function yields again a convex function.

As $F(\{\lambda_i\})$ is convex there is a unique maximum and any algorithm for multidimensional optimization can be used. Of course, it is possible to employ the generalized iterative scaling algorithm [8] used to train maximum entropy models to devise an optimization scheme for LLI. However, in this work, the simplex algorithm as described in [7] was used, as this is a robust general purpose method.

The parameters for the linear interpolation model are trained by the EM-algorithm. It is important to note, that both interpolation schemes have the same number of free parameters to be optimized, use the same ingredients and are language modeling tools to combine any set of models p_i.

4 Adaptation

This is the first section which describes a special application. The task is to build the best model suitable for a specific domain. To this end models trained on the background-corpus and models trained on the adaptation-corpus (which is typically from the target domain) are given. The background models are marked by the subscript "back" and the models from the target domain by the subscript "adap". We consider the combination of unigram-, bigram- and trigram-models. This yields

$$p(w|uv) = \frac{1}{Z(uv)} \quad p_{\mathrm{back}}(w|uv)^{\lambda_{A3}} \quad p_{\mathrm{adap}}(w|v)^{\lambda_{A2}} p_{\mathrm{back}}(w|v)^{\lambda_{B2}} \quad p_{\mathrm{adap}}(w)^{\lambda_{A1}} p_{\mathrm{back}}(w)^{\lambda_{B1}} \quad . \tag{6}$$

This model has five parameters which are determined on an additional cross-validation corpus also taken from the target domain.

The model (6) is related to FMA (fast marginal adaptation) that was developed in [5]. It is given by

$$p(w|uv) = \frac{1}{Z(uv)} \left(\frac{p_{\mathrm{adap}}(w)}{p_{\mathrm{back}}(w)} \right)^{\beta} p_{\mathrm{back}}(w|uv) \quad . \tag{7}$$

It was derived by using the first iteration of generalized iterative scaling (GIS). It is easily observed that FMA is a special case of (6) with $\lambda_{B1} = -\beta$, $\lambda_{A1} = \beta$, $\lambda_{B2} = 0$, $\lambda_{A2} = 0$, and $\lambda_{B3} = 1$.

5 Long Range Models

The second application will focus on combining models with different range. In [3, 9], bigrams and distance-2 bigrams (d2-bigrams) have been combined using maximum entropy methods to form a model for trigrams predicting word w after $u\,v$. Distance-2 bigrams are models $p_{\mathrm{d2}}(w|u)$ that ignore the word v immediately preceding w. In the framework of LLI this combination task is solved by

$$p(w|uv) = \frac{1}{Z(uv)} p(w)^{\lambda_U} p_{\mathrm{d1}}(w|v)^{\lambda_{d1}} p_{\mathrm{d2}}(w|u)^{\lambda_{d2}} \quad . \tag{8}$$

Here, $p_{\mathrm{d1}}(w|v)$ denotes the usual bigram. This model, may also be interpreted to result from a first iteration of GIS. The GIS-constraints are

$$\sum_u p(uvw) = p_{\mathrm{d1}}(vw) \tag{9}$$

and

$$\sum_v p(uvw) = p_{\mathrm{d2}}(uw) \quad . \tag{10}$$

As the initial model, the unigram is used. Hence, GIS gives a model with the same structure as (8) and additonal constraints for the interpolation weights: $\lambda_U = 1 - \gamma - \delta$, $\lambda_{d1} = \gamma$ and $\lambda_{d2} = \delta$ with $\gamma + \delta \leq 1$.

Of course, this scheme can be extended to combine distance-n bigrams and distance-$\{m, n\}$ trigrams to build a long-range M-gram.

6 Experiments

6.1 Data

For the adaptation experiments the Spoke 4 adaptation task from the 1994 ARPA evaluation was used [11]. Two different target domains are given, one with articles about Jackie Kennedy (denoted by "JK" in the tables) and one about Korea (denoted by "KO"'). For each domain, the size of the adaptation data is about 12 000 words. There are an additional 10 000 words for cross-validation and 2 000 for perplexity and recognition tests. The background models are trained on 240 million words of north American business news (NAB). The NAB-models where also used for experiments to combine models for different history.

The second set of data from the development set of the 1997 DARPA evaluation [12]. (transcripts of the acoustic training data: denoted by "TAT") consists of 380 000 words for training, 11 500 words for cross-validation and 11 500 words for testing For those texts an artificial vocabulary of 1 500 words was constructed that consists of words, phrases and fragments of words [10] and which has no unknown word.

6.2 Adaptation

Now, results for the adaptation task are reported in Tab. 1 and Tab. 2.

The resulting weights are given in Tab. 1 for FMA and for LLI, for using only unigram models and for using uni- and bigram models for adaptation. It is

Model	λ_{A1}	λ_{A2}	λ_{B1}	λ_{B2}	λ_{B3}
FMA JK	0.50	0.00	-0.50	0.00	1.00
FMA KO	0.42	0.00	-0.42	0.00	1.00
LLI Uni. JK	0.47	0.00	-0.44	0.00	0.91
LLI Uni. KO	0.43	0.00	-0.39	0.00	0.90
LLI Bi. JK	0.30	0.23	-0.42	-0.08	0.91
LLI Bi. KO	0.26	0.19	-0.37	0.00	0.86

Table 1: *Parameters for adaptation-task.*

interesting that increasing the number of free parameters to 3 (LLI Uni), not only effects λ_{A1} and λ_{B1} but also reduces λ_{B3} by 10%. When all five parameters are used, surprisingly λ_{B2} still vanishes and the weight of λ_{A1} is now distributed over λ_{A1} and λ_{A2}. For each model, the sum of all parameters is always 0.94 which hints at a constant importance of the zerogram model absorbed in the normalization.

Model	PP	WER
Trigram	198.4	23.2 %
FMA	160.1	22.6 %
Lin Uni.	175.2	22.3 %
LLI Uni.	152.7	21.6 %
Lin Bi.	148.8	21.8 %
LLI Bi.	143.0	21.6 %

Table 2: *Perplexity and WER for adaptation-task.*

In Tab. 2, the perplexity (PP) and word-error-rates (WER) results are reported for both target domains (JK and KO) combined. Going from FMA to a log-linear interpolation of the same component models (LLI Uni.) gives an improvement in perplexity of 5%. Including bigrams (LLI Bi.) gives an improvement of the same order. More important is the comparison of linear and log-linear interpolation. The improvements range from 13% to 4%. In all cases log-linear interpolation is the better way of combining models. The same is true for the word-error-rate.

6.3 Long Range Models

6.3.1 Bigrams and Distance-2 Bigrams

A simple and for practical applications important case is the combination of ordinary bigrams (d1-bigrams) and distance-2 bigrams to construct a model with an effective trigram context. Both NAB and TAT are used for the experiments. Hence, it is possible to compare the parameters of the two tasks. The numbers are depicted in Tab. 3. The parameters seem to be independent of the task, despite the fact that the two tasks differ. This robustness of the parameters is useful as it allows to use those as initial parameters for a new task. Only very few optimization steps will be necessary for a new task. Note that the parameters sum to approximately 1 as indicated by the derivation from GIS.

Model	λ_U	λ_{d1}	λ_{d2}	$\sum_i \lambda_i$
LLI NAB	-0.52	0.86	0.62	0.96
LLI TAT	-0.49	0.84	0.55	0.90

Table 3: *Parameters for effective-trigram-task.*

Model	PP	WER
Bigram	317.7	25.8 %
Lin. d1 + d2	302.1	25.8 %
LLI d1 + d2	250.1	25.0 %
Trigram	198.4	23.2 %

Table 4: *Perplexity and WER for the effective-trigram-task on NAB.*

The perplexity figures for NAB and TAT are given in Tab. 4 and Tab. 5 respectively. Linear interpolation of bigrams and d2-bigrams gives improvements between 1% and 5% whereas for log-linear interpolation the decrease in perplexity ranges between 19% and 21%. This is clearly the task for which log-linear interpolation outperforms the established linear interpolation. The WER-results for NAB are also given in Tab. 4. WER again goes down as perplexity is reduced.

6.3.2 Application to other long range models

Model	PP
Bigram	146.3
Trigram (M3)	99.7
Fourgram	97.8
Lin. d1 + d2	145.1
LLI d1 + d2	119.1
Lin. d1 + d2 + d3	145.1
LLI d1 + d2 + d3	115.8
Lin. M3 + d1 + d2 + d3	98.6
LLI M3 + d1 + d2 + d3	93.0

Table 5: *Perplexity on TAT.*

Of course, LLI is well suited to go beyond trigrams. In Tab. 5 two effective-fourgram models are investigated. The first one combines d1-, d2- and d3-bigrams. In this case, linear interpolation gives no additional improvement as the weight of the d3-bigram vanishes, because there is no component in the model that can compensate for unigram information also contained in the d3-bigrams. In contrast, for LLI there is an additional improvement of 3% because the unwanted parts

of the d3-bigram are compensated by an increase in the weight of the unigram. Finally, the trigram is added to the model (M3 + d1 + d2 + d3). This log-linearly combined model gives perplexities which are by 5% lower than the ones of the true fourgram.

7 Combination of Phrases and LLI

Phrases have shown nice improvements of bigram perplexity on WSJ. It is hence challenging to test how far one can get when combining LLI and phrases. The results on WSJ is summarized in Tab. 6. No training of the interpolation parameters has been done but they have been taken from the NAB tests as we also wanted to test the robustness of the parameters.

Model	PP
Words	
Bigram	102.1
Trigram	56.2
Words and Phrases	
Bigram	72.4
+ d2-Bi (Lin)	70.2
+ d2-Bi (LLI)	57.7

Table 6: *Combination with phrases: perplexity on WSJ.*

The original trigram had a perplexity of 56.2 and the first two steps outlined above (phrases and LLI) bring perplexity from an original 102.1 down to 57.7. Thus, we are now 2.5% above the original trigram.

8 Conclusion

We have presented a new general scheme for combining independently trained language models. Two applications have been discussed for which log-linear interpolation was always better than linear interpolation. In particular for combining models of different range the LLI is suitable.

Maximum-entropy (ME) models may also be used to combine different knowledge sources. However, given the component language models, LLI has far less free parameters (usually less than ten as compared to a few million for ME) which in addition are robust (i.e. do not vary much) when changing the task. Hence, training for LLI is far less an issue. What is still missing is a comparison for performance of LLI and ME. ME is expected to give better results but it is not clear whether the difference will be large enough to justify the effort of ME.

9 Acknowledgment

The author would like to thank Stefan Besling, Peter Beyerlein, and Jochen Peters for many stimulating discussions.

References

[1] F. Jelinek and R.L. Mercer: "Interpolated Estimation of Markov Source Parameters from Sparse Data", *Pattern Recognition in Practice*, pp. 381, 1980.

[2] R. Kneser and V. Steinbiss : "On the Dynamic Adaptation of Stochastic Language Models",*Proc. ICASSP*, pp. 586, 1993.

[3] R. Rosenfeld:"A Maximum Entropy Approach to Adaptive Language Modeling", *Computer Speech and Language*, pp. 187, 1996.

[4] J. Peters, internal reports.

[5] R. Kneser, J. Peters, and D. Klakow:"Language Model Adaptation using Dynamic Marginals", *EUROSPEECH*, pp. 1971, 1997.

[6] T. M. Cover and J. A. Thomas: "Elements of Information Theory", *John Wiley & Sons*, 1991.

[7] W.H. Press et al.:"Numerical Recipes", *Cambridge University Press*, 1989.

[8] J.N. Darroch and D. Ratcliff: "Generalized Iterative Scaling for Log-Linear Models", *The Annals of Mathematical Statistics*, pp. 1470 , 1972.

[9] M. Simons, H. Ney, and S.C. Martin: "Distant Bigram Language Modeling Using Maximum Entropy", *ICASSP*, pp. 787, 1997.

[10] D. Klakow: "Language-Model Optimization by Mapping of Corpora",*Proc. ICASSP*, accepted for publication, 1998.

[11] F. Kubala: "Design of the 1994 CSR Benchmark Tests.", *Proc. Spoken Language Systems Technology Workshop*, pp. 41, 1995.

[12] D. Graff: "The 1996 Broadcast News Speech and Language-Model Corpus",*DARPA Speech Recognition Workshop*, pp. 11, 1997.

[13] R. Iyer, M. Ostendorf: " Analyzing and Predicting Language Model Improvements", *ASRU*, pp. 254, 1997.