Language Model Adaptation for Tiny Adaptation Corpora

Dietrich Klakow

Spoken Language Systems
Saarland University, 66041 Saarbrücken, Germany
dietrich.klakow@lsv.uni-saarland.de

Abstract

In this paper we address the issue of building language models for very small training sets by adapting existing corpora. In particular we investigate methods that combine task specific unigrams with longer range models trained on a background corpus. We propose a new method to adapt class models and show how fast marginal adaptation can be improved. Instead of estimating the adaptation unigram only on the adaptation corpus, we study specific methods to adapt unigram models as well. In extensive experimental studies we show the effectiveness of the proposed methods. As compared to FMA as described in [1] we obtain an improvement of nearly 60% for ten utterances of adaptation data.

1. Introduction

Language model adaptation is the task of building a language model that is as good as possible given a domain specific adaptation corpus and a general background corpus. This is done to avoid a time consuming and expensive data collection for a new domain. In many transcription tasks it would be desirable to have language models specific to the users way of speaking and his topic. The focus of this paper is LM adaptation for a transcription task. However, also for the fast development and deployment of dialogue systems language model adaptation is important and past experience has shown that methods developed for language model adaptation for transcription tasks can be used for dialogue systems as well.

There have been many investigations on methods for language model adaptation. Bellegarda gave a very good overview of many recent techniques [2]. In this paper we would like to focus on methods that combine unigram models trained on the adaptation corpus and trigram models trained on the background corpus. As very often, a linear interpolation of word and class LMs is used, we develop a new method to also adapt class LMs by using domain specific emission probabilities. For adapting a word language model "fast marginal adaptation" (FMA) was proposed in [1]. We suggest a new adjusted version. For both, adapting the class and the word trigram, we need good unigrams. To this end, we also investigate methods to estimate good unigrams on adaptation corpora as small as ten utterances. A combination of fill-up [3] and document selection [4] proves to be very powerful. For fill-up we also use a new smoothing technique which gives a 10% improvement over the standard version.

The paper is organized as follows. First, adjusted FMA and the new class adaptation scheme are introduced. Next, methods to train good unigram models will be discussed. In the experimental section, we will show how to adapt a news paper corpus using a small fraction of the switchboard corpus [5].

2. Adapting Trigram Language Models

In the next section, we want to focus on using unigram language models to adapt trigrams. To capture the trigram dependencies in language, we can either adapt a class based trigram or a word trigram. All models use the smoothing technique proposed by R. Kneser and H. Ney.

2.1. Adapting Class Language Models

We would like to suggest a method to adapt class LMs. Class based language models have two components. An M-gram that predicts the next class given the classes of the words of the history and an emission model, that predicts a word given its class. Even for a moderate number of classes the class prediction model has quite a large number of parameters and can hence be best trained on a larger corpus for example the background corpus or a selection of relevant documents from the background corpus. However, the emission model is derived from a unigram. Therefore, all the techniques described in the previous section can be used to calculate an adapted emission model. The class model we use can be written as

$$P^C_{Adap}(w|h) = P_{Adap}(w|c)P_{BG}(c|c(w_{-1})...c(w_{M-1})) \quad (1)$$

where $P_{Adap}(w|c)$ is the emission model based on an adapted unigram, $h = w_{-1}...w_{M-1}$ is the M-gram history and c(w) is the class the word w belongs to.

There are many different possibilities to pick the classes. For the small ten utterances adaptation corpus, automatically training classes is infeasible and hence the classes were trained on the complete background corpus. We used one set of 1000 classes for all experiments, because this number gave overall the best performance.

2.2. Fast Marginal Adaptation

A different approach to combine unigram and trigram information is Fast Marginal Adaptation (FMA) proposed in [1]. It is based on one iteration of generalized iterative scaling (GIS) [7]. As an initial distribution, it uses the trigram trained on the background corpus. As a constraint, the desired trigram has to satisfy that its marginal is the unigram trained on the adaptation data. The first iteration of GIS based on these constraints yields

$$P_{Adap}(w|h) = \frac{1}{Z(h)}\left(\frac{P_{Adap}(w)}{P_{BG}(w)}\right)^{\beta} P_{BG}(w|h) \quad (2)$$

where β is a parameter derived from the weighting of the constraint equations. In our experiments, we found that $\beta = 0.9$ gives best performance. $Z(h)$ is a normalization. A very efficient way of calculating it is given in [1].

We can interpret this equation in the following way. The factor $P_{Adap}(w)/P_{BG}(w)$ scales certain words up or down that are more/less frequent in the adaptation data than in the background data. If the ratio is one nothing changes. This adds to the robustness of the method. Previous experiments have shown that FMA always outperforms linear interpolation. It was also checked in throughout this investigations and has been confirmed. The smaller the adaptation corpus, the larger the benefit of FMA over linear interpolation.

2.3. Adjusted Fast Marginal Adaptation

Starting from the interpretation of FMA, we can take a different perspective. We can use the unigram on the adaptation data $P_{Adap}(w)$ as the starting distribution and than adjust it with the ratio $P_{BG}(w|h)/P_{BG}(w)$ to include the prediction of the history. The formal derivation proceeds like the derivation of standard FMA using one iteration ogf GIS only that the roles of the various distribtuions are changed. This results in

$$P_{Adap}(w|h) = \frac{1}{Z(h)} \left(\frac{P_{BG}(w|h)}{P_{BG}(w)} \right)^{\beta} P_{Adap}(w) \qquad . \quad (3)$$

Words w that are more likely after a particular history h are now pushed up. In our experiments $\beta = 0.6$ gave optimal performance. The efficient calculation of the normalization $Z(h)$ can also be applied to this variant of FMA with minor modification in the algorithm given in [1]. As shown in the next section this adjusted version provides a consistent method over FMA even in combination with other adaptation schemes.

3. Adapting Unigram Language Models

In all previous studies, unigram models were trained on the adaptation corpus only. We will show how unigrams can be adapted as well to make best possible use of the above introduced adaptation schemes for trigrams. To this end, we have to address the issue how to build as good as possible domain specific unigrams. An experimental justification for this will be given in section 4.3.

In this section we review some existing methods, propose improvements and show how to build a good unigram by combination of these techniques using a small number of utterances from the Switchboard corpus for training.

3.1. Language Model Fill-Up

Besling and Meier proposed in [3] an adaptation strategy that the authors called “language model fill-up” or short “fill-up”. The key idea is to use the adaptation data whenever an M-gram is observed in the training data. In case of unseen M-grams in the adaptation data, the background model is used. For a unigram model fill-up can be mathematically formulated as

$$P(w) = \begin{cases} \dfrac{\mathrm{N_{Adap}}(w) - d}{\mathrm{N_{Adap}}} + \alpha P_{BG}(w) & \text{if } \mathrm{N_{Adap}}(w) > 0 \\[2ex] \alpha \cdot P_{BG}(w) & \text{else} \end{cases} \qquad (4)$$

where $N_{Adap}(w)$ is the frequency of word w in the adaptation data, d is the discounting parameter and $P_{BG}(w)$ a backing-off language model trained on the background data which may be either the full background corpus or a smaller subset determined by document selection as described in section (3.3).

3.2. Fill-Up with Continuous Absolute Discounting

As an extension of fill-up, we would like to propose to make the discounting parameter dependent on the count. From low counts you do not want to discount too much because they would experience a relatively sever modification but higher counts can also provide more probability mass for the backing off distribution without being changed to a value that is statistically significantly different from the original count. Specifically we use a rational interpolation formula

$$d(N) = \frac{d_0 + s(N-1)}{1 + g(N-1)} \qquad (5)$$

where d_0 is the usual absolute discounting parameter and s and g are additional new parameters. Typical parameters are $d_0 = 0.71$, $s = 1.23$ and $g = 0.13$, which results in $\frac{s}{g} = 9.5$. These parameters were optimized on development data. We also tried other formulas (e.g. substituting $N-1$ by $\sqrt{N-1}$) without achieving as good results.

3.3. Selecting Documents

In [4] we proposed to use the adaptation corpus as a “test set” and than try to find that subset of documents from the background corpus that minimizes perplexity on the adaptation corpus. The algorithm calculates the change in perplexity when a specific document is removed from the background corpus. In previous experiments we have shown, that it is not necessary to recalculate the change in perplexity for all remaining documents after one document has been removed. It is sufficient and also more robust to calculate the change in perplexity only once for all the documents and then remove all documents that decrease perplexity by more than a certain (possibly negative) amount. We applied this strategy to a unigram language model with absolute discounting. The resulting corpus of selected documents is used as the background corpus $P_{BG}(w)$ in fill-up.

4. Experiments

4.1. Data

For our experiment we used Switchboard [5] as the target domain. The full corpus consists of 2 million tokens. To enable adaptation experiments, we picked the first N utterances of the corpus as adaptation data. N was equally spaced on a logarithmic scale and varied from basically the complete corpus down to the first 10 utterances of switchboard. An utterance consists on average of 8.2. tokens. The first 10 utterances have 90 tokens including the “end of utterance” marker. Another ten utterances were used as the development set. The test set is the 1998 HUB-5 English Evaluation data as provided by LDC (42000 tokens).

As the background corpus we used the first 40 million words of the first CD of the North American News Text Corpus. We did only minor corpus processing. The document selection algorithm used the document markers as provided by LDC.

The document selection described in section 3.3 can also be used to determine an adapted vocabulary from a background corpus. Details of this procedure can be found in [4]. This results in a vocabulary of 64000 words and an OOV rate of 1.3% as compared to an OOV rate of 2.2% when determining the vocabulary from the Switchboard corpus alone.

4.2. Unigrams

In figure 1 a comparison of fill-up with standard absolute discounting (4) and the continuous absolute discounting (5) is shown. We can clearly observe a significant improvement for all sizes of the adaptation corpus. It is however largest for just the ten utterances of adaptation data. Here we observe an improve-

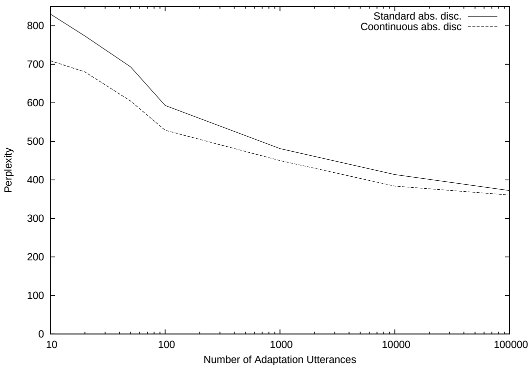

Figure 1: *Fill-up using standard absolute discounting and continuous absolute discounting for a varying number of utterances used for adaptation.*

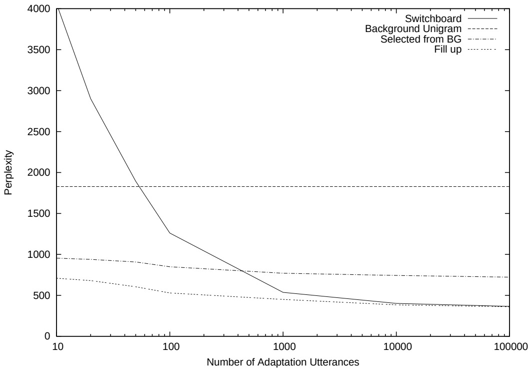

Figure 2: *Perplexity of the various unigram models.*

ment of 17%. Hence, we will only use this improved version of fill-up in all future experiments.

Figure 2 now compares the different strategies to train unigram models. We see that using the Switchboard specific data only results in a dramatic increase of perplexity as the amount of training data is decreased. At a certain point (50 utterances and less), it is even better to use the background corpus without any adaptation. Selecting documents from the background corpus reduces perplexity by about a factor of two. Only if we have more than 350 utterances using the original adaptation data is beneficial. Using the fill-up method with continuous absolute discounting results in a model that always outperforms the model trained on the adaptation data. For just ten utterances this model is better by a factor of 5.7 than the original one.

4.3. Influence of unigram perplexity on performance of adapted model

We still have to show that improving unigrams is worth the effort. To this end figure 3 shows the correlation of the perplexity of the unigram model and the adapted trigram models. We show results for adapted class based models and adjusted FMA, both without document selection. Each dot in the graph corresponds to a different unigram. All unigrams built during this investigations are shown. If the unigram perplexity is above 600, both types of models benefit from an improvement in the unigram component of the models. However, the adapted class models have the lower slope and hence and the points approach the diagonal as the unigram perplexity is decreased. Then the adapted class models follow the diagonal. This is due to the fact that the classes are trained on the background corpus and hence the predictive power of the mismatched classes is only of limited value. Training the classes instead on Switchboard removes the crossing. Decreasing the number of classes increases the slope and hence we also would get some improvement at lower unigram perplexities paying a price at higher unigram perplexities.

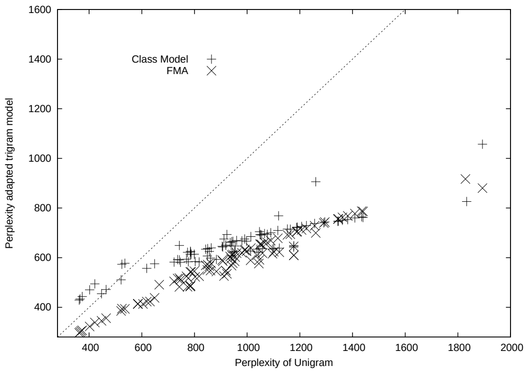

Figure 3: *Correlation of unigram perplexity and perplexity of adapted trigram models.*

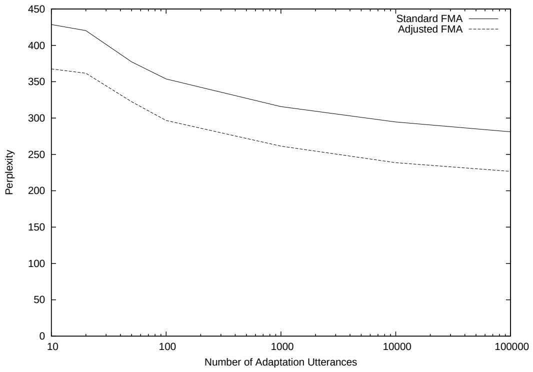

Figure 4: *Standard FMA and adjusted FMA*

Adjusted FMA has a larger slope and always benefits from improved unigrams. For bad unigrams, it is just as good as the adapted class models. For good unigrams, it never crosses the diagonal and hence is always able to add trigram information to the unigram.

4.4. Optimization of trigram models

In figure 4 we compare the standard variant of FMA with the adjusted version of FMA. For the unigram model $P_{Adap}(w)$ we used the fill-up with continuous absolute discounting. The background models were a unigram and a trigram trained on the complete background corpus. It turns out, that the adjusted version always outperforms the standard version. The relative improvement varies between 16% and 24%. This is probably due to the fact that $P_{Adap}(w)$ is the best among the three models integrated in FMA and the trigram information only "modifies" this model. Due to the improvement, we used the adjusted version for all other experiments.

We now want to study the influence of document selection on the performance of the trigram models. Also the pure word trigram benefits from article selection. This can be seen from figure 5. Using only 2.5% of the complete background corpus

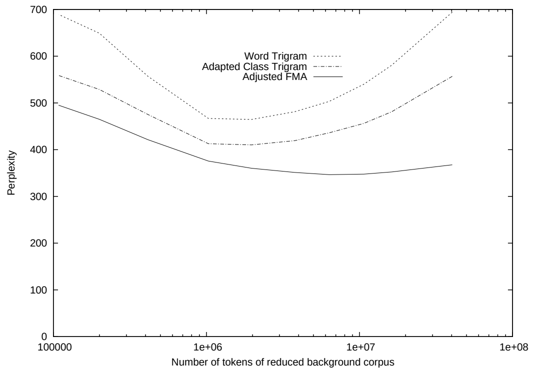

Figure 5: *Testing document selection with class adaptation and FMA. In this experiments we used ten utterances for adaptation.*

is a clear advantage. The same holds true for the adapted class models. However, due to the fact that the adapted class model uses a domain specific unigram it outperforms the word trigram. Adjusted FMA not only is better than the other two models, it can also make use of a larger background corpus. There is some small improvement when the 75% irrelevant documents are removed from the corpus, but the improvement is minor. The benefit is rather the reduced size of the model.

4.5. Comparison of different trigram adaptation schemes

In this final experimental subsection a discussion of the various trigram models will be done. The results are summarized in figure 6. Like in the case of the unigram models, just using the adaptation corpus to build a trigram gives extremely large perplexities for small adaptation corpora. For just ten utterances, the perplexity is 4328 and well outside the graph. Using a class trigram or a word trigram trained on the background corpus already gives better results for ten utterances, which is not really surprising. Adapting the class model reduces perplexity by a factor of 1.9 (as compared to the non adapted class model) for ten utterances. Adjusted FMA as the corresponding method for the word model reduces perplexity also by a factor of 1.9, as compared to a word based trigram on the background corpus. Finally a linear interpolation of the two models is done. For ten utterances this results in a perplexity of 273 which is a reduction of 35% as compared to adjusted FMA. When the complete switchboard corpus is available the perplexity of the word trigram is 139, which is of course still much smaller. However, figure 6 also gives a feeling for how much additional adaptation utterances give how much additional perplexity reduction. For the development of transcription systems as well as dialogue systems this curve captures the trade-off of performance versus effort for data collection. For practical applications one would also have to make sure that the adaptation utterances are carefully chosen in particular if the number of adaptation utterances is small. In this respect the experiments shown here are a pessimistic estimate, because the first N utterances for switchboard were used. The effect of this can be best seen when looking at the data points for $N = 10$ and $N = 20$ utterances. There is nearly no decrease in perplexity because the utterances 11-20 from switchboard are strongly correlated with the utterances 1-10 and hence carry no new information about the target domain.

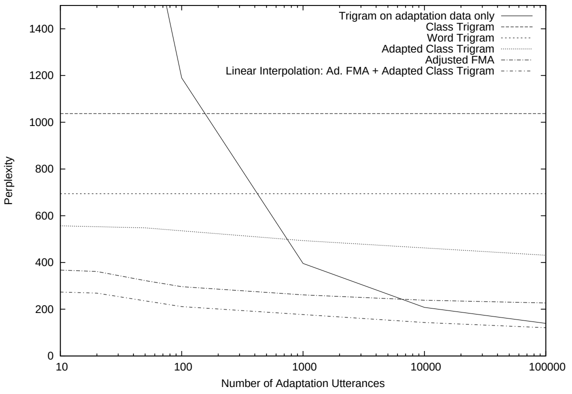

Figure 6: *Perplexity of adapted trigram models.*

5. Conclusions and Future Work

This paper investigates methods to combine adapted unigram language models with trigram language models trained on a background corpus. The proposed adaptation scheme for class models as well as the adjusted version of FMA prove to be effective. To make best possible use of these methods we also study the adaptation of unigram models instead of just training unigrams on the adaptation corpus. Experiments on adapting a news paper corpus to switchboard have shown that language models with reasonable perplexity can be built even for ten utterances of adaptation data.

Future work will use a much large background corpus, possibly the gigaword corpus and also investigate in more detail the variations that come from fluctuations in a small adaptation corpus.

6. Acknowledgments

The author would like Andreas Merkel for interesting discussions. This work was partially funded by the SmartWeb project under contract 01IMD01M.

7. References

[1] Kneser, R., Peters J. and Klakow, D., “Language Model Adaptation Using Dynamic Marginals”, Proc. EUROSPEECH, 4:1971-1974, 1997.

[2] Bellegarda, J.R., “Statistical Language Model Adaptation: Review and Perspectives”, in Speech Communication, Special Issue on Adaptation Methods for Speech Recognition, J.C. Junqua and C.J. Wellekens, Eds., 42:93-108, 2004.

[3] Besling, S. and Meier, H.G., “Language model speaker adaptation”, Proc. EUROSPEECH, 1995.

[4] Klakow, D., “Selecting Articles from the Language Model Training Corpus”, Proc. ICASSP, 3:1695-1699, 2000.

[5] Godfrey, J.J. Holliman, E.C. McDaniel, J., “SWITCHBOARD: telephone speech corpus for research and development”, Proc. ICASSP, 1:517-520, 1992.

[6] Moore, G. and Young S., “Class-based language model adaptation using mixtures of word-class weights” Proc. ICSLP, 2000.

[7] Darroch, J.N. and Ratcliff , D., “Generalized Iterative Scaling for Log-Linear Models”, The Annals of Mathematical Statistics, 1470, 1972.