Language model adaptation with MAP estimation and the perceptron algorithm

Michiel Bacchiani, Brian Roark and Murat Saraclar
AT&T Labs-Research, 180 Park Ave., Florham Park, NJ 07932, USA
{michiel,roark,murat}@research.att.com

Abstract

In this paper, we contrast two language model adaptation approaches: MAP estimation and the perceptron algorithm. Used in isolation, we show that MAP estimation outperforms the latter approach, for reasons which argue for combining the two approaches. When combined, the resulting system provides a 0.7 percent absolute reduction in word error rate over MAP estimation alone. In addition, we demonstrate that, in a multi-pass recognition scenario, it is better to use the perceptron algorithm on early pass word lattices, since the improved error rate improves acoustic model adaptation.

1 Introduction

Most common approaches to language model adaptation, such as count merging and model interpolation, are special cases of maximum a posteriori (MAP) estimation (Bacchiani and Roark, 2003). In essence, these approaches involve beginning from a smoothed language model trained on out-of-domain observations, and adjusting the model parameters based on in-domain observations. The approach ensures convergence, in the limit, to the maximum likelihood model of the in-domain observations. The more in-domain observations, the less the out-of-domain model is relied upon. In this approach, the main idea is to change the out-of-domain model parameters to match the in-domain distribution.

Another approach to language model adaptation would be to change model parameters to correct the errors made by the out-of-domain model on the in-domain data through discriminative training. In such an approach, the baseline recognizer would be used to recognize in-domain utterances, and the parameters of the model adjusted to minimize recognition errors. Discriminative training has been used for language modeling, using various estimation techniques (Stolcke and Weintraub, 1998; Roark et al., 2004), but language model adaptation to novel domains is a particularly attractive scenario for discriminative training, for reasons we discuss next.

A key requirement for discriminative modeling approaches is training data produced under conditions that are close to testing conditions. For example, (Roark et al., 2004) showed that excluding an utterance from the language model training corpus of the baseline model used to recognize that utterance is essential to getting word error rate (WER) improvements with the perceptron algorithm in the Switchboard domain. In that paper, 28 different language models were built, each omitting one of 28 sections, for use in generating word lattices for the omitted section. Without removing the section, no benefit was had from models built with the perceptron algorithm; with removal, the approach yielded a solid improvement. More time consuming is controlling acoustic model training. For a task such as Switchboard, on which the above citation was evaluated, acoustic model estimation is expensive. Hence building multiple models, omitting various subsections is a substantial undertaking, especially when discriminative estimation techniques are used.

Language model adaptation to a new domain, however, can dramatically simplify the issue of controlling the baseline model for producing discriminative training data, since the in-domain training data is not used for building the baseline models. The purpose of this paper is to compare a particular discriminative approach, the perceptron algorithm, which has been successfully applied in the Switchboard domain, with MAP estimation, for adapting a language model to a novel domain. In addition, since the MAP and perceptron approaches optimize different objectives, we investigate the benefit from combination of these approaches within a multi-pass recognition system.

The task that we focus upon, adaptation of a general voicemail recognition language model to a customer service domain, has been shown to benefit greatly from MAP estimation (Bacchiani and Roark, 2003). It is an attractive test for studying language model adaptation, since the out-of-domain acoustic model is matched to the new domain, and the domain shift does not raise the OOV rate significantly. Using 17 hours of in-domain observations, versus 100 hours of out-of-domain utterances, (Bacchiani and Roark, 2003) reported a reduction in WER from 28.0% using the baseline system to 20.3%

with the best performing MAP adapted model. In this paper, our best scenario, which uses MAP adaptation and the perceptron algorithm in combination, achieves an additional 0.7% reduction, to 19.6% WER.

The rest of the paper is structured as follows. In the next section, we provide a brief background for both MAP estimation and the perceptron algorithm. This is followed by an experimental results section, in which we present the performance of each approach in isolation, as well as several ways of combining them.

2 Background

2.1 MAP language model adaptation

To build an adapted n-gram model, we use a count merging approach, much as presented in (Bacchiani and Roark, 2003), which is shown to be a special case of maximum a posteriori (MAP) adaptation. Let $\mathbf{w_O}$ be the out-of-domain corpus, and $\mathbf{w_I}$ be the in-domain sample. Let h represent an n-gram history of zero or more words. Let $c_k(hw)$ denote the raw count of an n-gram hw in $\mathbf{w_k}$, for $k \in \{O, I\}$. Let $\hat{p}_k(hw)$ denote the standard Katz backoff model estimate of hw given $\mathbf{w_k}$. We define the corrected count of an n-gram hw as:

$$\hat{c}_k(hw) \quad = \quad |\mathbf{w_k}|\, \hat{p}_k(hw) \tag{1}$$

where $|\mathbf{w_k}|$ denotes the size of the sample $\mathbf{w_k}$. Then:

$$\tilde{p}(w \mid h) = \frac{\tau_h \hat{c}_O(hw) + \hat{c}_I(hw)}{\tau_h \sum_{w'} \hat{c}_O(hw') + \sum_{w'} \hat{c}_I(hw')} \tag{2}$$

where τ_h is a state dependent parameter that dictates how much the out-of-domain prior counts should be relied upon. The model is then defined as:

$$p^*(w \mid h) = \begin{cases} \tilde{p}(w \mid h) & \text{if } c_O(hw) + c_I(hw) > 0 \\ \alpha\, p^*(w \mid h') & \text{otherwise} \end{cases} \tag{3}$$

where α is the backoff weight and h' the backoff history for history h.

The principal difficulty in MAP adaptation of this sort is determining the mixing parameters τ_h in Eq. 2. Following (Bacchiani and Roark, 2003), we chose a single mixing parameter for each model that we built, i.e. $\tau_h = \tau$ for all states h in the model.

2.2 Perceptron algorithm

Our discriminative n-gram model training approach uses the perceptron algorithm, as presented in (Roark et al., 2004), which follows the general approach presented in (Collins, 2002). For brevity, we present the algorithm, not in full generality, but for the specific case of n-gram model training.

The training set consists of N weighted word lattices produced by the baseline recognizer, and a gold-standard transcription for each of the N lattices. Following (Roark et al., 2004), we use the lowest WER hypothesis in the lattice as the gold-standard, rather than the reference transcription. The perceptron model is a linear model with k feature weights, all of which are initialized to 0. The algorithm is incremental, i.e. the parameters are updated at each example utterance in the training set in turn, and the updated parameters are used for the next utterance. After each pass over the training set, the model is evaluated on a held-out set, and the best performing model on this held-out set is the model used for testing.

For a given path π in a weighted word lattice $\mathcal{L}$, let $w[\pi]$ be the cost of that path as given by the baseline recognizer. Let $\mathcal{G}_\mathcal{L}$ be the gold-standard transcription for $\mathcal{L}$. Let $\Phi(\pi)$ be the K-dimensional feature vector for π, which contains the count within the path π of each feature. In our case, these are unigram, bigram and trigram feature counts. Let $\bar{\alpha}_t \in \mathbb{R}^K$ be the K-dimensional feature weight vector of the perceptron model at time t. The perceptron model feature weights are updated as follows

1. For the example lattice $\mathcal{L}$ at time t, find $\hat{\pi}_t$ such that

$$\hat{\pi}_t \quad = \quad \operatorname*{argmin}_{\pi \in \mathcal{L}} \left(w[\pi] + \lambda \Phi(\pi) \cdot \bar{\alpha}_t\right) \tag{4}$$

   where λ is a scaling constant.

2. For the $0 \leq k \leq K$ features in the feature weight vector $\bar{\alpha}_t$,

$$\bar{\alpha}_{t+1}[k] \quad = \quad \bar{\alpha}_t[k] + \Phi(\hat{\pi}_t)[k] - \Phi(\mathcal{G}_\mathcal{L})[k] \tag{5}$$

   Note that if $\hat{\pi}_t = \mathcal{G}_\mathcal{L}$, then the features are left unchanged.

As shown in (Roark et al., 2004), the perceptron feature weight vector can be encoded in a deterministic weighted finite state automaton (FSA), so that much of the feature weight update involves basic FSA operations, making the training relatively efficient in practice. As suggested in (Collins, 2002), we use the averaged perceptron when applying the model to held-out or test data. After each pass over the training data, the averaged perceptron model is output as a weighted FSA, which can be used by intersecting with a lattice output from the baseline system.

3 Experimental Results

We evaluated the language model adaptation algorithms by measuring the transcription accuracy of an adapted voicemail transcription system on voicemail messages received at a customer care line of a telecommunications network center. The initial voicemail system, named Scanmail, was trained on general voicemail messages collected from the mailboxes of people at our research

site in Florham Park, NJ. The target domain is also composed of voicemail messages, but for a mailbox that receives messages from customer care agents regarding network outages. In contrast to the general voicemail messages from the training corpus of the Scanmail system, the messages from the target domain, named SSNIFR, will be focused solely on network related problems. It contains frequent mention of various network related acronyms and trouble ticket numbers, rarely (if at all) found in the training corpus of the Scanmail system.

To evaluate the transcription accuracy, we used a multi-pass speech recognition system that employs various unsupervised speaker and channel normalization techniques. An initial search pass produces word-lattice output that is used as the grammar in subsequent search passes. The system is almost identical to the one described in detail in (Bacchiani, 2001). The main differences in terms of the acoustic model of the system are the use of linear discriminant analysis features; use of a 100 hour training set as opposed to a 60 hour training set; and the modeling of the speaker gender which in this system is identical to that described in (Woodland and Hain, 1998). Note that the acoustic model is appropriate for either domain as the messages are collected on a voicemail system of the same type. This parallels the experiments in (Lamel et al., 2002), where the focus was on AM adaptation in the case where the LM was deemed appropriate for either domain.

The language model of the Scanmail system is a Katz backoff trigram, trained on hand-transcribed messages of approximately 100 hours of voicemail (1 million words). The model contains 13460 unigram, 175777 bigram, and 495629 trigram probabilities. The lexicon of the Scanmail system contains 13460 words and was compiled from all the unique words found in the 100 hours of transcripts of the Scanmail training set.

For every experiment, we report the accuracy of the one-best transcripts obtained at 2 stages of the recognition process: after the first pass lattice construction (FP), and after vocal tract length normalization and gender modeling (VTLN), Constrained Model-space Adaptation (CMA), and Maximum Likelihood Linear regression adaptation (MLLR). Results after FP will be denoted FP; results after VTLN, CMA and MLLR will be denoted MP.

For the SSNIFR domain we have available a 1 hour manually transcribed test set (10819 words) and approximately 17 hours of manually transcribed adaptation data (163343 words). In all experiments, the vocabulary of the system is left unchanged. Generally, for a domain shift this can raise the error rate significantly due to an increase in the OOV rate. However, this increase in error rate is limited in these experiments, because the majority of the new domain-dependent vocabulary are acronyms which are covered by the Scanmail vocabulary through individual letters. The OOV rate of the SSNIFR test set, using the Scanmail vocabulary is 2%.

System	FP	MP
Baseline	32.7	28.0
MAP estimation	23.7	20.3
Perceptron (FP)	26.8	23.0
Perceptron (MP)	–	23.9

Table 1: Recognition on the 1 hour SSNIFR test set using systems obtained by supervised LM adaptation on the 17 hour adaptation set using the two methods, versus the baseline out-of-domain system.

Following (Bacchiani and Roark, 2003), τ_h in Eq. 2 is set to 0.2 for all reported MAP estimation trials. Following (Roark et al., 2004), λ in Eq. 4 is also (coincidentally) set to 0.2 for all reported perceptron trials. For the perceptron algorithm, approximately 10 percent of the training data is reserved as a held-out set, for deciding when to stop the algorithm.

Table 1 shows the results using MAP estimation and the perceptron algorithm independently. For the perceptron algorithm, the baseline Scanmail system was used to produce the word lattices used in estimating the feature weights. There are two ways to do this. One is to use the lattices produced after FP; the other is to use the lattices produced after MP.

These results show two things. First, MAP estimation on its own is clearly better than the perceptron algorithm on its own. Since the MAP model is used in the initial search pass that produces the lattices, it can consider all possible hypotheses. In contrast, the perceptron algorithm is limited to the hypotheses available in the lattice produced with the unadapted model.

Second, training the perceptron model on FP lattices and applying that perceptron at each decoding step outperformed training on MP lattices and only applying the perceptron on that decoding step. This demonstrates the benefit of better transcripts for the unsupervised adaptation steps.

The benefit of MAP adaptation that leads to its superior performance in Table 1 suggests a hybrid approach, that uses MAP estimation to ensure that good hypotheses are present in the lattices, and the perceptron algorithm to further reduce the WER. Within the multi-pass recognition approach, several scenarios could be considered to implement this combination. We investigate two here.

For each scenario, we split the 17 hour adaptation set into four roughly equi-sized sets. In a first scenario, we produced a MAP estimated model on the first 4.25 hour subset, and produced word lattices on the other three subsets, for use with the perceptron algorithm. Table 2 shows

System	MAP Pct.	FP	MP
Baseline	0	32.7	28.0
MAP estimation	100	23.7	20.3
MAP estimation	25	25.6	21.5
Perceptron (FP)	25	23.8	20.5
Perceptron (MP)	25	–	20.8

Table 2: Recognition on the 1 hour SSNIFR test set using systems obtained by supervised LM adaptation on the 17 hour adaptation set using the first method of combination of the two methods, versus the baseline out-of-domain system.

the results for this training scenario.

A second scenario involves making use of all of the adaptation data for both MAP estimation and the perceptron algorithm. As a result, it requires a more complicated control of the baseline models used for producing the word lattices for perceptron training. For each of the four sub-sections of the adaptation data, we produced a baseline MAP estimated model using the other three subsections. Using these models, we produced training lattices for the perceptron algorithm for the entire adaptation data set. At test time, we used the MAP estimated model trained on the entire adaptation set, as well as the perceptron model trained on the entire set. The results for this training scenario are shown in table 3.

Both of these hybrid training scenarios demonstrate a small improvement by using the perceptron algorithm on FP lattices rather than MP lattices. Closely matching the testing condition for perceptron training is important: applying a perceptron trained on MP lattices to FP lattices hurts performance. Iterative training did not produce further improvements: training a perceptron on MP lattices produced by using both MAP estimation and a perceptron trained on FP lattices, achieved no improvement over the 19.6 percent WER shown above.

4 Discussion

This paper has presented a series of experimental results that compare using MAP estimation for language model domain adaptation to a discriminative modeling approach for correcting errors produced by an out-of-domain model when applied to the novel domain. Because the MAP estimation produces a model that is used during first pass search, it has an advantage over the perceptron algorithm, which simply re-weights paths already in the word lattice. In support of this argument, we showed that, by using a subset of the in-domain adaptation data for MAP estimation, and the rest for use in the perceptron algorithm, we achieved results at nearly the same level as MAP estimation on the entire adaptation set.

System	MAP Pct.	FP	MP
Baseline	0	32.7	28.0
MAP estimation	100	23.7	20.3
Perceptron (FP)	100	22.9	19.6
Perceptron (MP)	100	–	19.9

Table 3: Recognition on the 1 hour SSNIFR test set using systems obtained by supervised LM adaptation on the 17 hour adaptation set using the second method of combination of the two methods, versus the baseline out-of-domain system.

With a more complicated training scenario, which used all of the in-domain adaptation data for both methods jointly, we were able to improve WER over MAP estimation alone by 0.7 percent, for a total improvement over the baseline of 8.4 percent.

Studying the various options for incorporating the perceptron algorithm within the multi-pass rescoring framework, our results show that there is a benefit from incorporating the perceptron at an early search pass, as it produces more accurate transcripts for unsupervised adaptation. Furthermore, it is important to closely match testing conditions for perceptron training.

References

Michiel Bacchiani and Brian Roark. 2003. Unsupervised language model adaptation. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 224–227.

Michiel Bacchiani. 2001. Automatic transcription of voicemail at AT&T. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*.

Michael Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1–8.

L. Lamel, J.-L. Gauvain, and G. Adda. 2002. Unsupervised acoustic model training. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 877–880.

Brian Roark, Murat Saraclar, and Michael Collins. 2004. Corrective language modeling for large vocabulary ASR with the perceptron algorithm. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*.

A. Stolcke and M. Weintraub. 1998. Discriminitive language modeling. In *Proceedings of the 9th* H*ub-5 Conversational Speech Recog nition Workshop*.

P.C. Woodland and T. Hain. 1998. The September 1998 HTK Hub 5E System. In *The Proceedings of the* 9^{th} *Hub-5 Conversational Speech Recognition Workshop*.