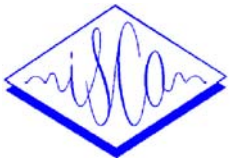

INTERSPEECH 2004 - ICSLP
8th International Conference on Spoken Language Processing
ICC Jeju, Jeju Island, Korea
October 4-8, 2004

Unsupervised Language Model Adaptation Methods for Spontaneous Speech

Luc Lussier, Edward W.D. Whittaker, Sadaoki Furui

Department of Computer Science
Tokyo Institute of Technology
2-12-1, Ookayama, Meguro-ku
Tokyo 152-8552 Japan
{llussier,edw,furui}@furui.cs.titech.ac.jp

Abstract

In this paper we examine the performance of three different unsupervised language model adaptation schemes applied to speech recognition of spontaneous speech lecture presentations. Two of the schemes have been described previously in the literature while the third is a variation of one of the other two schemes. All three schemes are based on a combination of word n-gram and class n-gram models and use an initial transcription hypothesis to adapt the parameters of the class model. In each case the adapted class model is linearly interpolated with the baseline word n-gram model and the combination is then applied in a subsequent recognition step. One of the schemes described also contains an element of domain adaptation in which the transcription hypothesis is also used to determine the interpolation weights of several class models each of which is built on automatically derived clusters of presentations. We also investigate multi-pass adaptation for each scheme and show this gives additional improvements in performance. Relative improvements in word error rate of up to 12.5% (3.4% absolute) are obtained on a held-out test set with the best adaptation scheme.

1. Introduction

The performance of state-of-the art automatic speech recognition systems has steadily improved as both new research and a larger amount of data are applied to this challenging task. This is especially true regarding the difficult task of spontaneous speech recognition where the availability of the "Corpus of Spontaneous Japanese" (CSJ)[11] has provided a considerable amount of relevant training data. Using a state-of-the-art word-based language model our current baseline word error rate for the CSJ corpus averaged over the three test sets described by Kawahara et al. [5] is 26.8%. In this paper, we investigate three different unsupervised language model adaptation schemes applied to speech recognition of test data in the CSJ corpus.

There has been continual and extensive interest in developing unsupervised language model adaptation schemes for a variety of tasks and domains. Two recent approaches that improve performance with varying degrees of success are described in [1, 12]. In [13] two different methods for unsupervised language model adaptation are examined for the same task we consider in this paper, although the test sets and baselines do not permit a quantitative comparison.

Two of the schemes we examine in this paper have been reported recently in [17, 18] and [8]. The third scheme is a simple, more constrained, variation on the scheme described in [17, 18]. All three adaptation schemes are based on a combination of word n-gram and class n-gram models and use an initial transcription hypothesis to adapt the parameters of the class model. In each adaptation scheme the adapted class model is linearly interpolated with the baseline word n-gram model and the combination employed in a subsequent recognition step. One of the adaptation schemes described also contains an element of domain adaptation in which the transcription hypothesis is also used to determine the interpolation weights of several class models each of which is built on automatically derived clusters of presentations.

In Section 2 we describe in detail the adaptation schemes examined in the paper. In Section 3 we describe the data used for training and testing and the experimental setup. Results for the different adaptation schemes are then presented in Section 4 and a discussion and conclusion given in Sections 5 and 6.

2. Language model adaptation

The language models used in our experiments are based on the combination of a general, word-based language model and one or more specialized, class-based language models using linear interpolation as illustrated by the following:

$$p\left(w|h\right) = \lambda_0 \cdot p_g\left(w|h, T_0\right) + \sum_{m=1}^{M} \lambda_m \cdot p_{s_m}\left(w|h, N, W\right), \quad (1)$$

where w is the current word for which the probability is calculated, h is the history, λ_m is the weight assigned to each model such that $\sum \lambda_m = 1$ ($\lambda_m > 0$), $p_g(\cdot)$ is the general (word n-gram) language model built using data from the whole training corpus T_0, and $p_s(\cdot)$ corresponds to one of M specialized language models. The specialised model is always a class n-gram model, for which we use two parameters to describe the data source (N) for training the class n-gram component, and the data source (W) used to train the word-given-class unigram component as follows:

$$p_s\left(w \mid h, N, W\right) = p(w \mid C(w), W) \cdot p(C(w) \mid C(h), N).$$

We use the notation $(N|W)$ to describe each type of adaptation scheme in terms of the source of training data for each component. In this paper we investigate three different adaptation schemes with the following source combinations: $(H|H)$, $(T_0|H)$ and $(T_j|H)$, where H is the transcription hypothesis from the speech recognizer output, T_0 is the whole training data and T_j is a partition of the training data obtained by clustering presentations in T_0 into J presentation clusters i.e. $M = J$ in

Equation (1). For the $(H|H)$ and $(T_0|H)$ schemes, no partitioning of the training corpus is used, thus $M = 1$. The adaptation scheme in [17, 18] is described by $(H|H)$.

For all adaptation schemes, the word-class definitions $C(w)$ are trained on the whole training set T_0. The word-class definition is built using the word clustering algorithm described by Kneser and Ney [6] to create $|C|$ different word classes where each word is a member of only one class such that $C_i \cap C_j = \varnothing \quad \forall i, j, i \neq j$.

2.1. Clustering presentations for the $(T_j|H)$ adaptation scheme

For the $(T_j|H)$ adaptation scheme we require a partition of the training data T_0 into J clusters of similar presentations T_j. Each cluster T_j, where $1 < j \leq J$, contains a certain number of presentations and each presentation is a member of a single cluster such that $T_i \cap T_j = \varnothing \quad \forall i, j, i \neq j$. The entire corpus of training presentations is referred to as T_0.

The clustering method used is a bottom-up, agglomerative type of clustering based on a word co-occurrence metric. It was used in [4, 12] and is based on [14]. The clustering process works according to the following algorithm:

- Place each presentation P in a single cluster.
- Iterate, until only one cluster is left:
  - For each pair of presentation clusters P_i and P_j, compute the similarity metric S_{ij}.
  - Merge the two clusters that have the highest similarity.

To determine how similar two presentation clusters are, the following similarity metric S_{ij} is used:

$$S_{ij} = \sum_{w \in P_i \cap P_j} \frac{N_{ij}}{|P^w|} \times \frac{1}{|P_i| \times |P_j|} \tag{2}$$

where P_i and P_j represent two presentation clusters, $|P^w|$is the number of presentation clusters that contain the word w, $|P_i|$ is the number of unique words in the cluster P_i and N_{ij} is defined as follows:

$$N_{ij} = \sqrt{\frac{N_i + N_j}{N_i \times N_j}} \tag{3}$$

where N_i, which represents the number of presentations in the cluster, is a normalization factor used to prevent the development of a single large cluster.

The clustering is based on all the words from each presentation and the sequence of merge operations is preserved so that any desired number of clusters can be obtained.

2.2. Multi-pass adaptation

We also investigate multi-pass adaptation using each of the three adaptation schemes described. The first pass in each case, corresponds to recognition using the baseline n-gram language model with no adaptation. Each subsequent pass takes the transcription hypothesis from the previous pass for building the adapted model. It should be noted that, in our experiments, an entirely new recognition pass is performed for each pass rather than lattice rescoring being used. However, we believe it is unlikely that this has a significantly positive or negative effect on performance.

3. Experimental setup

We perform recognition experiments using the Julius speech recognition engine version 3.3p3 developed by Lee et al. [7]. In order to accommodate various combinations of word and word-class models, Julius was slightly modified such that language model probabilities could be obtained from an external library.

3.1. Acoustic model

The acoustic features used for the experiments are 25 dimensional vectors consisting of 12 MFCCs, their delta as well as the delta log energy. All the models used are gender dependent triphone HMMs with 3000 shared states and 16 Gaussian mixtures. Cepstral mean subtraction is also applied to each utterance.

Table 1 shows the number of presentations and how many hours are used to train the acoustic models. The academic only models are used for test sets 1 and 2, and models containing both academic and extemporaneous presentations are used for test set 3.

Model	# talks (# hours)	
	Female	Male
Academic only	166 (42)	787 (186)
Academic and extemporaneous	988 (176)	1508 (310)

Table 1: Summary of the data used to create the acoustic models.

3.2. Baseline language model

The baseline language model is built from the transcribed content of about 2590 presentations providing almost 7.5 million words of training data with a vocabulary size of 30678 words. Because there are generally no spaces between characters in written Japanese, the concept of a word boundary is not clearly defined. Thus, as defined by Shinozaki and Furui [15], a word refers to a Japanese morpheme, that is an arbitrary number of characters, extracted by a morphological analyzer developed by Uchimoto et al. [16] for the CSJ corpus.

All of the training data was used to build a baseline forward word bigram and a baseline reverse word trigram as needed by the Julius speech recognition engine. A variation of the smoothing technique developed by Kneser and Ney introduced in [3] is used with all language models.

All the language models used are built with an extended set of tools originally based on the CMU-Cambridge language modelling toolkit [2].

3.3. Development and evaluation sets

The first of the three test sets defined in the CSJ benchmark paper by Kawahara et al. [5] is used as a development set and the last two as evaluation sets. Each test set contains 10 presentations. Test sets 1 and 2 contain only academic presentations while test set 3 comprises only extemporaneous presentations. In addition, test set 1 contains only presentations made by male speakers whereas test sets 2 and 3 contain both female and male speakers in equal proportion. The number of words in test sets 1 and 2 are similar but test set 3 is smaller, containing slightly less than 66% of either test set 1 or 2. These statistics are summarized in Table 2.

Test set	Number of words	Speech style
1 (dev)	26515	A,M
2	26923	A,M,F
3	17213	E,M,F

Table 2: Total number of words per test set and style of data: A=academic, E=extemporaneous, M=male, F=female.

4. Results

For the $(H|H)$ adaptation scheme we found that, on the development set (test set 1), the optimal interpolation coefficient was $\lambda_1 = 0.3$ and the optimal number of word-classes $\mid C \mid = 130$. These values agree closely with the values reported previously in [17, 18]. We use these values for both the $(H|H)$ adaptation scheme and also for the $(T_0|H)$ scheme. For the adaptation scheme using clustered presentations $(T_j|H)$ it was found in [8] that $\mid T \mid = 8$ and $\mid C \mid = 514$ gave optimal performance on test set 1. For this adaptation scheme, the interpolation weights $\lambda_{0,\ldots,M}$ are computed using the EM algorithm so as to maximise the likelihood of the transcription hypothesis. However, for this weight estimation procedure the word-given-class component was trained using T_0 rather than H (i.e. effectively a $(T_j|T_0)$ adaptation scheme is used) to prevent over-fitting to the errorful transcription hypothesis and to avoid obtaining unreliable interpolation weights. For each adaptation scheme the appropriate forward bigram and reverse trigram models are built as required by the Julius recognizer.

The word error rate performance of all three adaptation schemes for three adaptation passes on the development set (test set 1) is given in Table 3. For comparison we also show the performance of supervised adaptation of the model when the correct transcription is used (supervised).

Pass	$(H\|H)$	$(T_0\|H)$	$(T_j\|H)$
1 (baseline)	26.7	26.7	26.7
2	25.2	24.9	24.8
3	25.1	24.1	24.3
4	25.1	24.3	24.4
Supervised	16.9	24.4	23.6

Table 3: Word error rate results (%) for each adaptation scheme and multiple passes on test set 1 together with results for supervised adaptation.

The results in Table 3 show that up to two adaptation passes can be performed while still obtaining robust improvements in performance. Additional passes produce results that are not significantly different. For the evaluation experiments we therefore only execute three passes in total for each adaptation scheme. The word error rate results are presented in Tables 4 and 5 for test sets 2 and 3 respectively.

Pass	$(H\|H)$	$(T_0\|H)$	$(T_j\|H)$
1 (baseline)	27.1	27.1	27.1
2	24.8	24.8	24.3
3	24.6	24.2	23.7

Table 4: Word error rate results (%) for each adaptation scheme and multiple passes on test set 2.

Pass	$(H\|H)$	$(T_0\|H)$	$(T_j\|H)$
1 (baseline)	25.8	25.8	25.8
2	24.5	24.8	24.5
3	24.4	24.4	24.5

Table 5: Word error rate results (%) for each adaptation scheme and multiple passes on test set 3.

5. Discussion

One of the surprising things we note from the results in Table 3 is that for the $(T_0|H)$ and $(T_j|H)$ adaptation schemes we achieve unsupervised adaptation performance similar to that from supervised adaptation[1], in which the correct transcription is used for adaptation. This suggests that for these particular adaptation scheme configurations we have approached the limit in performance for unsupervised adaptation. Clearly, however, we might expect to obtain further improvements if we could robustly estimate other parameters such as the interpolation weights in an unsupervised manner.

From the results, we also note that each additional pass gives a reliable reduction in word error rate on test set 2 though has little effect on test set 3. The inclusion of an element of domain adaptation in the $(T_j|H)$ scheme also gives substantial improvements on test set 2, where it gives a word error rate 0.9% abs. lower than with the $(H|H)$ scheme. However, similar, large improvements are not obtained on test set 3. There is no clear reason for these differences except that the baseline language model for test set 3 is better trained since there is more extemporaneous training data available than academic. Another likely explanation is that each presentation in test set 3 is approximately two-thirds the length of the presentations in test sets 1 and 2. This means there is substantially less data available for adaptation; the results agree with observations made in [8] when fractional parts of presentations were used for adaptation.

The most reliable information we can use for unsupervised adaptation is the unigram statistics. Higher-order information such as bigrams and trigrams is slightly less than twice and three times more unreliable, respectively (the probability of there being an error in any given n-gram increases as $1-(1-x)^n$ where x is the word error rate). In this respect the $(H|H)$ scheme can be seen as the most unconstrained of the three adaptation schemes we investigated since it incorporates both unigram and n-gram ($n = 2, 3$) statistics from the transcription hypothesis. It appears however that using the class definitions trained on T_0 is sufficient to ensure that any unreliable higher-order adaptation data does not significantly reduce performance. The $(T_0|H)$ adaptation scheme, on the other hand, does not permit any higher-order adaptation statistics to be used in the adapted model and its results are as good as or better than the $(H|H)$ scheme. This suggests that removing the influence of higher-order adaptation information is indeed beneficial for this particular adaptation scheme and, moreover, that the $(H|H)$ scheme does not in fact reliably incorporate higher-order adaptation information as may have been previously assumed.

One related factor affecting adaptation performance that has not yet been investigated is that of the baseline word error rate.

[1]The reason why the supervised result for the $(T_0|H)$ adaptation scheme is worse (though not statistically significantly) than the unsupervised result is attributed to the use of smoothing and the fixed interpolation weight which was not optimised separately for supervised adaptation.

It might be expected that at a certain (higher) word error rate, adaptation actually worsens performance. Similarly, at lower word error rates we might expect to obtain correspondingly larger improvements from multiple passes or from the inclusion of higher-order statistics. In these experiments we have chosen a development set that is sufficiently similar to at least one of the test sets which avoids such problems to some extent.

6. Conclusion

In this paper we have compared three different unsupervised adaptation schemes. We showed that up to two adaptation passes can improve performance significantly for the adaptation schemes we examined. Performance was confirmed to be dependent on how much data was available for adaptation and also on how well trained the baseline model was. Although there was no clear best adaptation scheme overall, the $(T_j|H)$ scheme gave consistently good results on all test sets and the best result (12.5% relative improvement in word error rate) on test set 2, the characteristics of which are most similar to those of the development set.

Primarily we have only used unigram information in the three adaptation schemes since these comprise the most reliable statistics in the hypothesised transcription. Indeed we showed that, with these particular adaptation schemes, the incorporation of higher-order information does not help and sometimes even worsens performance. However, by using word lattices from the recognizer, rather than the single best hypothesis, we might expect to obtain both more reliable unigram and higher-order statistics. This would allow us to reduce the constraints of our current approaches and hopefully obtain even greater improvements in performance. This is expected to be a focus of future work.

7. Acknowledgments

The authors would like to thank Koji Iwano, Takahiro Shinozaki and Tadasuke Yokoyama for useful discussions that contributed substantially to the work presented in this paper; Takahiro Shinozaki also provided the acoustic models used in our experiments.

8. References

[1] Michiel Bacchiani and Brian Roark Unsupervised Language Model Adaptation In *Proceedings ICASSP*, pp. 224–227, 2003.

[2] Philip R. Clarkson and Ronald Rosenfeld. Statistical Language Modeling using the CMU-Cambridge Toolkit. In *Proceedings EUROSPEECH*, pp. 2707–2710, 1997.

[3] Joshua T. Goodman. A Bit of Progress in Language Modeling. Technical report, Microsoft Research, 2001.

[4] Rukmini Iyer and Mari Ostendorf. Modelling Long Distance Dependence in Language: Topic Mixtures vs. Dynamic Cache Models. In *Proceedings ICSLP*, pp. 236–239, 1996.

[5] Tatsuya Kawahara, Hiroaki Nanjo, Takahiro Shinozaki, and Sadaoki Furui. Benchmark Test for Speech Recognition using the Corpus of Spontaneous Japanese. In *Proceedings SSPR*, pp. 135–138, 2003.

[6] Reinhard Kneser and Hermann Ney. Improved Clustering Techniques for Class-based Statistical Language Modelling. In *Proceedings EUROSPEECH*, pp. 973–976, 1993.

[7] A. Lee, T. Kawahara, and K. Shikano. Julius — an Open Source Real-time Large Vocabulary Recognition Engine. In *Proceedings EUROSPEECH*, pp. 1691–1694, 2001.

[8] Luc Lussier, Edward W. D. Whittaker, and Sadaoki Furui. Combinations of Language Model Adaptation Methods applied to Spontaneous Speech. Proceedings of the Spontaneous Speech Science and Technology Workshop, February 2004.

[9] Luc Lussier, Edward W. D. Whittaker, and Sadaoki Furui. Looking at Alternatives within the Framework of n-gram Based Language Modeling for Spontaneous Speech Recognition. IEICE SP2003-141, pp. 169–174, December 2003.

[10] Luc Lussier, Edward W. D. Whittaker, and Sadaoki Furui. Word-class Models for Unsupervised Language Model Adaptation applied to Spontaneous Speech Recognition. Acoustical Society of Japan, Spring Meeting, March 2004.

[11] K. Maekawa, H. Koiso, Sadaoki Furui, and H. Isahara. Spontaneous Speech Corpus of Japanese. In *Proceedings of LREC*, pp. 947–952, 2000.

[12] Gareth Moore and Steve Young. Class-based Language Model Adaptation using Mixtures of Word-class Weights. In *Proceedings ICSLP*, pp. 512–515, 2000.

[13] T.R. Niesler and D. Willett. Unsupervised Language Model Adaptation for Lecture Speech Transcription In *Proceedings ICSLP*, pp. 1413–1416, 2002.

[14] S. Sekine. Automatic Sublanguage Identification for a New Text. In *Second Annual Workshop on Very Large Corpora*, pages 109–120, Kyoto, Japan, 1994.

[15] Takahiro Shinozaki and Sadaoki Furui. Analysis on Individual Differences in Automatic Transcription of Spontaneous Presentations. In *Proceedings ICASSP*, pp. 729–732, 2002.

[16] Kiyotaka Uchimoto, Chikashi Nobata, Atsushi Yamada, Satoshi Sekine, and Hitoshi Isahara. Morphological Analysis of the Corpus of Spontaneous Japanese. In *Proceedings SSPR*, pp. 159–162, Tokyo, Japan, 2003.

[17] Tadasuke Yokoyama, Takahiro Shinozaki, Koji Iwano, and Sadaoki Furui. Unsupervised Class-based Language Model Adaptation for Spontaneous Speech Recognition. In *Proceedings ICASSP*, pp. 236–239, 2003.

[18] Tadasuke Yokoyama, Takahiro Shinozaki, Koji Iwano, and Sadaoki Furui. Unsupervised Language Model Adaptation using Word Classes for Spontaneous Speech Recognition. In *Proceedings SSPR*, pp. 71–74, Tokyo, Japan, 2003.