CAMBRIDGE UNIVERSITY

ENGINEERING DEPARTMENT

USE OF CONTEXTS IN LANGUAGE MODEL INTERPOLATION AND ADAPTATION

X. Liu, M. J. F. Gales & P. C. Woodland

CUED/F-INFENG/TR.630

Feb 2009

Cambridge University Engineering Department
Trumpington Street
Cambridge. CB2 1PZ
England

E-mail: xl207@eng.cam.ac.uk
http://mi.eng.cam.ac.uk/~xl207

Abstract

Language models (LMs) are often constructed by building multiple component models that are combined using interpolation weights. By tuning these interpolation weights, using either perplexity or discriminative approaches, it is possible to adapt LMs to a particular task. This paper investigates the use of context dependent interpolation in both interpolation and test-time self-adaptation of language models. Depending on the previous word contexts, a discrete *history weighting function* is used to dynamically adjust the contribution from each component model. Under this framework, minimum Bayes risk (MBR) based discriminative training schemes are also proposed. As this dramatically increases the number of parameters to estimate, robust weight estimation schemes are required. Several approaches are described in this paper. The first approach uses training data to ensure robust estimation. An important issue with this method is to handle the bias to corpus size. An inverse corpus size weighted version of perplexity, *normalized perplexity*, is proposed. The second is based on MAP estimation where interpolation weights of lower order contexts, for example, are used as smoothing priors. The third approach uses class history interpolation weights. Rather than using standard Ney's alogorithm to derive the word-to-class mapping, a scheme specifically aimed at class contexts based weights is proposed. For unseen contexts an efficient weight back-off scheme is also used. A range of schemes to combine context dependent weights obtained from training and test data to improve LM adaptation are also proposed. Consistent perplexity and error rate gains of 6% relative were obtained on a state-of-the-art broadcast audio recognition task.

Contents

Table of Notations

$P(w_i \| h_i^{n-1})$	n-gram probability of the i^{th} word given a $i-1$ word history context h_i^{n-1}
λ	context free, global language model interpolation weight vector
$\tilde{\lambda}$	current estimate of global interpolation weight vector
$\hat{\lambda}$	optimal estimate of global interpolation weight vector
$\boldsymbol{\phi}(h)$	context dependent weight vector for history h
$g(w)$	class mapping for word w
$\mathcal{F}$	training criterion
$\mathcal{C}(h)$	sufficient weight statistics for history context h
$\mathcal{O}$	speech utterance with finite length
$\mathcal{W}$	arbitrary word sequence
$\mathcal{W}_{\mathsf{ref}}$	reference word sequence
$\mathcal{L}(\mathcal{W}, \mathcal{W}_{\mathsf{ref}})$	error loss a word sequence $\mathcal{W}$ against a reference $\mathcal{W}$
τ	smoothing constant used in MAP estimation
D	smoothing constant used in EBW update
E	smoothing constant used in EBW update

1 Introduction

A crucial component in automatic speech recognition (ASR) systems is the language model. Back-off n-gram models remain the dominant language modeling approach for state-of-art ASR systems [14]. In these systems language models are often constructed by combining information from a collection of diverse data sources. In order to more effectively use multiple sources, a model level combination is often preferred to a simple merge of source specific statistics. Under this framework, component n-gram models are first trained using individual text corpora prior to a probability interpolation. To reduce the mismatch between the interpolated model and target domain of interest, the interpolation weights may be tuned by minimizing the perplexity on some held-out data similar to the target domain. These weights indicate the "usefulness" of each source for a particular task. To further improve robustness to varying styles or tasks, unsupervised test-set adaptation to a particular broadcast show, for example, may be used. As directly adapting n-gram word probabilities is impractical on limited amounts of data, standard adaptation schemes only involve updating the global interpolation weights.

There are two major issues with this standard adaptation scheme. First, the diversity among data sources manifests itself in a wide range of factors, including source of collection, epoch, genre, modeling resolution and robustness, topics and styles. The precise nature of each source is jointly determined by a combination of these factors. Some of them may be sufficiently modeled on a higher level using global, context *independent* weights, such as source of collection, epoch and genre. Others factors, such as n-gram modeling resolution and generalization, topics and styles, can affect the contribution of sources on a local, context *dependent* basis. Thus the usefulness of a particular source for a domain may vary depending on the word context for both model tuning and adaptation. Using global weights take no account of this local variability. Hence, it is preferable to increase the modeling resolution of weight parameters by adding contextual information [3, 12, 21]. Second, the correlation between perplexity and error rate is well known to be fairly weak for current ASR systems. Hence, it may be useful to use discriminative training techniques [6, 17, 19, 31, 18, 28, 4, 20]. These schemes do not make incorrect modeling assumption and explicitly aim at reducing the underlying error rate cost function. In particular the minimum Bayes risk (MBR) criterion provides a flexible framework that can generalize to a wide range of error cost functions [15, 29, 8].

To address these issues, this paper investigates the use of context dependent interpolation in both training and test-time self-adaptation of language models. Under this framework MBR based discriminative training schemes are also proposed. As this dramatically increases the number of parameters to estimate, robust weight estimation schemes are required. Several approaches are described in this paper. The first approach uses training data to ensure robust estimation for a general form of context dependent LM interpolation. An important issue with this method is to handle the bias to corpus size. In this work an inverse corpus size weighted version of perplexity, *normalized perplexity*, is proposed. The second is based on MAP estimation where either interpolation weights of lower order contexts are used as priors, or those that are estimated on the training data. The third approach uses class history interpolation weights. Rather than using standard approaches to derive the word-to-class mapping, a scheme specifically aimed at class contexts based weights is proposed. For unseen contexts an efficient weight back-off scheme is also used. A variety of methods to integrate context dependent weights in model interpolation and adaptation are proposed. Performance of context dependent interpolation and adaptation is evaluated on a state-of-the-art Mandarin Chinese broadcast transcription task. Consistent perplexity and error rate improvements were obtained over baseline systems using global, context free weights. An overall relative improvement of 6% in CER was achieved over an unadapted baseline with global LM interpolation weights tuned on test set reference.

2 Language Model Interpolation and Adaptation

A common approach for LM adaptation is to adjust the global linear interpolation weights for a mixture model. For word based n-gram models, the log probability of the L word sequence $\mathcal{W} =< w_1, w_2, ..., w_i, ..., w_L >$, is given by

$$\ln P(\mathcal{W}) \quad = \quad \sum_{i=1}^{L} \ln P(w_i|h_i^{n-1}) \qquad (1)$$

where w_i denote the ith word of $\mathcal{W}$, and h_i^{n-1} represents its n-gram history of a maximum length of $n-1$ words if available, $< w_{i-1}, w_{i-2}, ..., w_{i-n+1} >$. For language modeling in current ASR systems, two forms of probability interpolation are available: a linear or log-linear interpolation of component models. These two forms in turn are instances of mixtures of experts (MoE) [34] and products of experts (PoE) [11].

2.1 Linear and Log-linear Interpolation

The linearly interpolated word probability is computed as,

$$P(w_i|h_i^{n-1}) = \sum_m \lambda_m P_m(w_i|h_i^{n-1}) \tag{2}$$

where λ_m is the global weight for the m^{th} component model. A comparable log-linear interpolation is given by,

$$P(w_i|h_i^{n-1}) = \frac{1}{Z_{h_i^{n-1}}} \exp\left(\sum_m \lambda_m \log P_m(w_i|h_i^{n-1})\right) \tag{3}$$

where $Z_{h_i^{n-1}}$ is a normalization term to ensure the interpolated probability to be a valid distribution. These may be ignored when considered under a discriminative framework as in *Maximum entropy* models and *logistic regression* [7, 28]. However, the exact computation of the normalization term for log-linear models, or PoE models in general, is non-trivial. Analytical solutions may be available only for certain forms of density functions. In this paper linear interpolation will be focused on.

2.2 Estimation of Interpolation Weights

Assuming "sufficient" amount of training data is available, and a strong correlation between perplexity and error rate exists, a perplexity, or equivalently maximum likelihood, based estimation schemes is often used for optimizing global interpolation weights. Alternatively, it is possible to use discriminative approaches, such as minimum Bayes risk (MBR), to estimate the weights.

Perplexity/ML Estimation: If the global interpolation weights are often found by minimizing the **perplexity** (PP) measure,

$$\mathcal{F}_{\mathsf{PP}} = \exp\left\{-\frac{\ln P(\mathcal{W})}{L}\right\} \tag{4}$$

it is equivalent to maximizing the log-likelihood of the entire word sequence, $\ln P(\mathcal{W})$, as in equations 1 and 2. $\mathcal{W}$ is the held-out data data for interpolation tuning, or supervision in case of model adaptation. For simplicity, tuning of the interpolated model is taken as an example here. The optimal linear interpolation weight for the mth component model can be derived by,

$$\hat{\lambda}_m^{\mathsf{ML}} = \arg\max_{\lambda_m} \{\ln P(\mathcal{W})\} \tag{5}$$

Under a constraint such that $0 < \lambda_m < 1$ and $\sum_m \lambda_m = 1$, the Baum-Welch (BW) algorithm may be used to iteratively re-estimate the weights,

$$\hat{\lambda}_m^{\mathsf{ML}} = \frac{\mathcal{C}_m^{\mathsf{ML}}(\mathsf{null})}{\sum_m \mathcal{C}_m^{\mathsf{ML}}(\mathsf{null})} \tag{6}$$

where the ML context independent statistics, $\mathcal{C}_m^{\mathsf{ML}}(\mathsf{null})$,

$$\mathcal{C}_m^{\mathsf{ML}}(\mathsf{null}) = \tilde{\lambda}_m \frac{\partial \ln P(\mathcal{W})}{\partial \lambda_m}\bigg|_{\lambda=\tilde{\lambda}} \tag{7}$$

$\tilde{\lambda}$ is the current weight estimate, and the derivative against weight parameters,

$$\frac{\partial \ln P(\mathcal{W})}{\partial \lambda_m} = \sum_{i=1}^{L} \frac{P_m(w_i|h_i^{n-1})}{\sum_m \lambda_m P_m(w_i|h_i^{n-1})} \tag{8}$$

If perplexity base adaptation is performed in supervised mode the correct transcription is required.

MBR Estimation: The minimum Bayes risk (MBR) criterion is expressed as the expected recognition error of an ASR system on a sequence of speech observations, $\mathcal{O}$. It is computed by summing over the cost function contribution from all possible hypotheses $\{\mathcal{W}\}$, weighted by their posterior probabilities, $P(\mathcal{W}|\mathcal{O})$. The global weight parameters are optimized by [15, 29, 8],

$$\begin{aligned}\hat{\lambda}_m^{\mathsf{MBR}} &= \arg\min_{\lambda_m} \{\mathcal{F}_{\mathsf{MBR}}(\mathcal{O})\} \\ &= \arg\min_{\lambda_m} \left\{\sum_{\mathcal{W}} P(\mathcal{W}|\mathcal{O})\mathcal{L}(\mathcal{W},\mathcal{W}_{\mathsf{ref}})\right\} \end{aligned} \tag{9}$$

where $\mathcal{L}(\mathcal{W},\mathcal{W}_{\mathsf{ref}})$ denotes the defined recognition error rate measure of hypothesis $\mathcal{W}$ against the reference hypothesis $\mathcal{W}_{\mathsf{ref}}$. A variety of forms of cost function, such as word or character level error rates, may be used depending on the underlying evaluation metric being considered. This provides more flexibility, compared with other discriminative criteria, such as maximum mutual information (MMI), as the loss function is not necessarily restricted to one particular form. By definition if $\mathcal{W}_{\mathsf{ref}}$ is the correct transcription MBR adaptation will be performed in supervised mode.

Numerical methods may be used to optimize the MBR criterion. However, these schemes can be slow and difficult to guarantee convergence. The Extended Baum-Welch (EBW) algorithm provides an efficient iterative optimization scheme for a family of rational objective functions, including MBR [10]. For global linear interpolation weights under a sum-to-one constraint, the re-estimation formula is given by,

$$\hat{\lambda}_m^{\mathsf{MBR}} = \frac{\mathcal{C}_m^{\mathsf{MBR}}(\mathsf{null})}{\sum_m \mathcal{C}_m^{\mathsf{MBR}}(\mathsf{null})} \tag{10}$$

where the discriminative context independent statistics, $\mathcal{C}_m^{\mathsf{MBR}}(\mathsf{null})$, are computed as

$$\mathcal{C}_m^{\mathsf{MBR}}(\mathsf{null}) = \tilde{\lambda}_m \left.\frac{\partial \mathcal{F}_{\mathsf{MBR}}(\mathcal{O})}{\partial \lambda_m}\right|_{\lambda=\tilde{\lambda}} + D \tag{11}$$

$\tilde{\lambda}$ is the current weight estimate, and D a tunable regularization constant controlling the convergence speed. Following the MBR criterion given in equation 9, the partial derivative in the above may be re-expressed as [20],

$$\frac{\partial \mathcal{F}_{\mathsf{MBR}}(\mathcal{O})}{\partial \lambda_m} = \sum_{\mathcal{W}} \frac{\partial P(\mathcal{W}|\mathcal{O})\mathcal{L}(\mathcal{W},\mathcal{W}_{\mathsf{ref}})}{\partial \ln p(\mathcal{O},\mathcal{W})} \frac{\partial \ln p(\mathcal{O},\mathcal{W})}{\partial \lambda_m} \tag{12}$$

where the first term can be derived as the following,

$$\frac{\partial P(\mathcal{W}|\mathcal{O})\mathcal{L}(\mathcal{W},\mathcal{W}_{\mathsf{ref}})}{\partial \ln p(\mathcal{O},\mathcal{W})} = P(\mathcal{W}|\mathcal{O})\left[1 - P(\mathcal{W}|\mathcal{O})\right]\mathcal{L}(\mathcal{W},\mathcal{W}_{\mathsf{ref}}) \tag{13}$$

The second term in equation 12 is independent of the acoustic model distribution $p(\mathcal{O}|\mathcal{W})$, and effectively identical to the sufficient statistics required by perplexity based weights optimization given in equation 8.

3 Context Dependent Interpolation and Adaptation

As discussed, the global weights assigned to component n-gram language models take no account of the surrounding contexts. In order to incorporate more context information, a more general form is to introduce a context dependent *history weighting function*, $\boldsymbol{\phi}(h)$. Using an n-gram context history, the interpolated word probability in equation 2 thus becomes,

$$P(w_i|h_i^{n-1}) = \sum_m \phi_m(h_i^{n-1}) P_m(w_i|h_i^{n-1}) \tag{14}$$

where $\boldsymbol{\phi}_m(h_i^{n-1})$ is the m^{th} component weight vector for n-gram history h_i^{n-1}. The same Markov chain assumption of component n-gram models is made such that the interpolation weights for word w_i only depends on the preceding $n-1$ words. These context dependent weights are also constrained to be positive and sum-to-one. A history weighting function can bear either a discrete or continuous form. Each has its own advantages and disadvantages, as will be discussed in the rest of this section.

3.1 Discrete History Weighting Functions

Discrete history weighting functions are effectively look-up tables of n-gram context dependent weights, where each distinct history may have its own weight vector. They can be represented by a tree structured hierarchy of context dependent interpolation weights. An example is shown in figure 1 for back-off tri-gram LMs. Such hierarchy will be extensively used in the rest of this paper. Such tree structure of contexts allows an efficient back-off strategy to be used when word or class contexts that are not seen in the training or adaptation data,

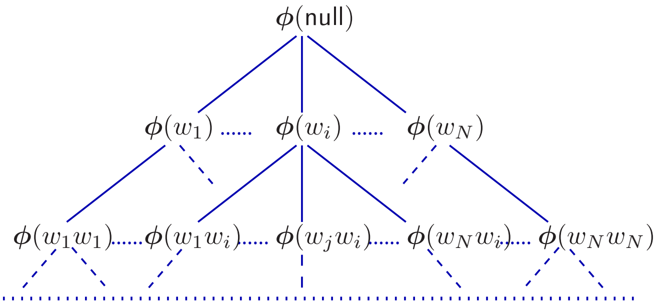

$\boldsymbol{\phi}(\mathsf{null})$
$\boldsymbol{\phi}(w_1)$
$\boldsymbol{\phi}(w_i)$
$\boldsymbol{\phi}(w_N)$
$\boldsymbol{\phi}(w_1w_1)$
$\boldsymbol{\phi}(w_1w_i)$
$\boldsymbol{\phi}(w_jw_i)$
$\boldsymbol{\phi}(w_Nw_i)$
$\boldsymbol{\phi}(w_Nw_N)$

Figure 1: Hierarchy of context dependent interpolation weights for tri-gram back-off n-gram language models s with a history of two words maximum.

$$\boldsymbol{\phi}^{\mathsf{bo}}(h_i^{n-1}) \quad = \quad \begin{cases} \boldsymbol{\phi}(h_i^{n-1}) & \text{if } \exists \ \boldsymbol{\phi}(h_i^{n-1}) \\ \boldsymbol{\phi}(h_i^{n-2}) & \text{else if } \exists \ \boldsymbol{\phi}(h_i^{n-2}) \\ \ldots & \ldots \\ \boldsymbol{\phi}(\mathsf{null}) & \text{otherwise} \end{cases} \tag{15}$$

The above back-off recursion may eventually simplify to the global, context independent weight, $\boldsymbol{\phi}(\mathsf{null})$, if the shortest history based on the most immediate preceding word, w_{i-1}, is unavailable. In contrast to standard word or class based n-gram models, no normalization term is required. These discrete weight parameters can be either maximum or discriminatively trained using the BW or EBW algorithm as discussed in section 2, except that context dependent statistics $\{\mathcal{C}_m(h_i^{n-1})\}$ are now required in update formulae of equations 6 and 10. As the number of weight parameters to estimate increases exponentially with context length, robust weight estimation schemes are required when limited amount of training data is available.

In common with n-gram models, the above form of context dependent interpolation weights with back-off may be viewed by a probabilistic finite state model for each component LM. One common approach to represent these models is to use *log semi-ring* based weighted finite state transducers (WFSTs) [24, 25, 26, 27]. A weighted transducer associates weights such as probabilities, durations, penalties, or any other quantity that accumulates linearly along paths within a graph, to each pair of input and output symbol sequences. Many types of modeling information used in speech recognition systems, such as n-gram models and the context dependent weight models considered here, involve a stochastic finite-state mappings between symbol sequences. WFSTs provide a generic and well-defined framework to represent them.

Now it is interesting to use WFST representation to re-examine the difference between context free LM interpolation given in equation (2), and the context dependent form given in equation (14). In both cases, the WFST representation of a linearly interpolated LM may be derived using a component level *composition* between the n-gram and interpolation weight transducers prior to a final *union* operation. This is given by,

$$L_G \quad = \quad \left(L_G^{(1)} \circ L_{\boldsymbol{\phi}}^{(1)}\right) \cup \ldots \cup \left(L_G^{(m)} \circ L_{\boldsymbol{\phi}}^{(m)}\right) \cup \ldots \cup \left(L_G^{(M)} \circ L_{\boldsymbol{\phi}}^{(M)}\right) \tag{16}$$

where $L_G^{(m)}$ is the n-gram model transducer and $L_{\boldsymbol{\phi}}^{(m)}$ the interpolation weight transducer for the m^{th} component. The difference between context free and dependent LM interpolation lies in the precise nature of interpolation weight information being represented in the interpolation weight transducers, $\{L_{\boldsymbol{\phi}}^{(m)}\}$.

Consider two simple back-off bi-gram language models that are trained on two different text sources for a three word vocabulary $\{w_1, w_2, w_3\}$. Let the sentence start and end tokens be represented by word w_1 and w_3 respectively. These two component bi-gram model sets, together with the corresponding transducers, $L_G^{(1)}$ and $L_G^{(2)}$, are shown in figure 2(a) and 2(b). In both component LM transducers, n-gram log probabilities appear as the negated arc weights. The 1-gram back-off weights are represented by special non-emitting arcs that have no associated output symbol, as marked with "< epsilon >" in the figure. The WFST representation of the context free interpolation weights for the two component models, 0.3 and 0.7 (1.220 and 0.360 as negated log) are shown in the two simple transducers of figure 2(c) and 2(d) respectively. After a linear interpolation via the WFST operation given in equation 16, the resulting transducer is shown in figure 2(e). It can be shown that context free, global interpolation simply increases the costs of all non-back-off arc within each component n-gram model sub-transducer by its own weight. In figure 2(e), the interpolated probability of any n-gram is represented by the marginalization over the probability of all partial paths, $\{l\}$, in the transducer that leave from a given history context h_i^{n-1} and outputs a particular word symbol w_i. This is given by,

$$P(w_i|h_i^{n-1}) \quad = \sum_{l:h_i^{n-1}\mapsto w_i} \exp(-\omega(l)) \tag{17}$$

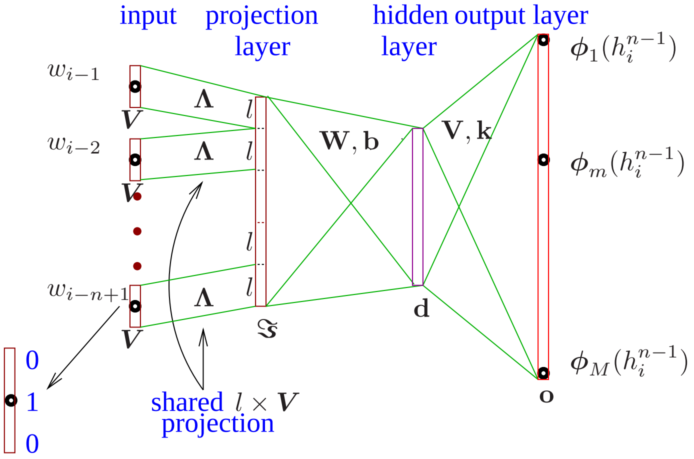
where $\omega(l)$ is the total cost of partial path l. Note that for clarity, the two component n-gram transducers are left un-joined at the terminal nodes after union.

The previously discussed limitation of context independent interpolation is also illustrated in figure 2. For bi-gram $P(w_3|w_2)$, the first component LM gives a bi-gram log-probability of -8.5, as is shown by the arc from node 3 to 4 in figure 2(a). A lower score of -15.5 is assigned by the second component LM via a back-off to 1-gram $P(w_3)$, as is shown by the back-off arc from node 3 to 1, and the 1-gram arc from node 1 to 4 in figure 2(b). For this context the probability contribution from the two component language models clearly contradicts the assignment of context free, global interpolation weights of 0.3 and 0.7.

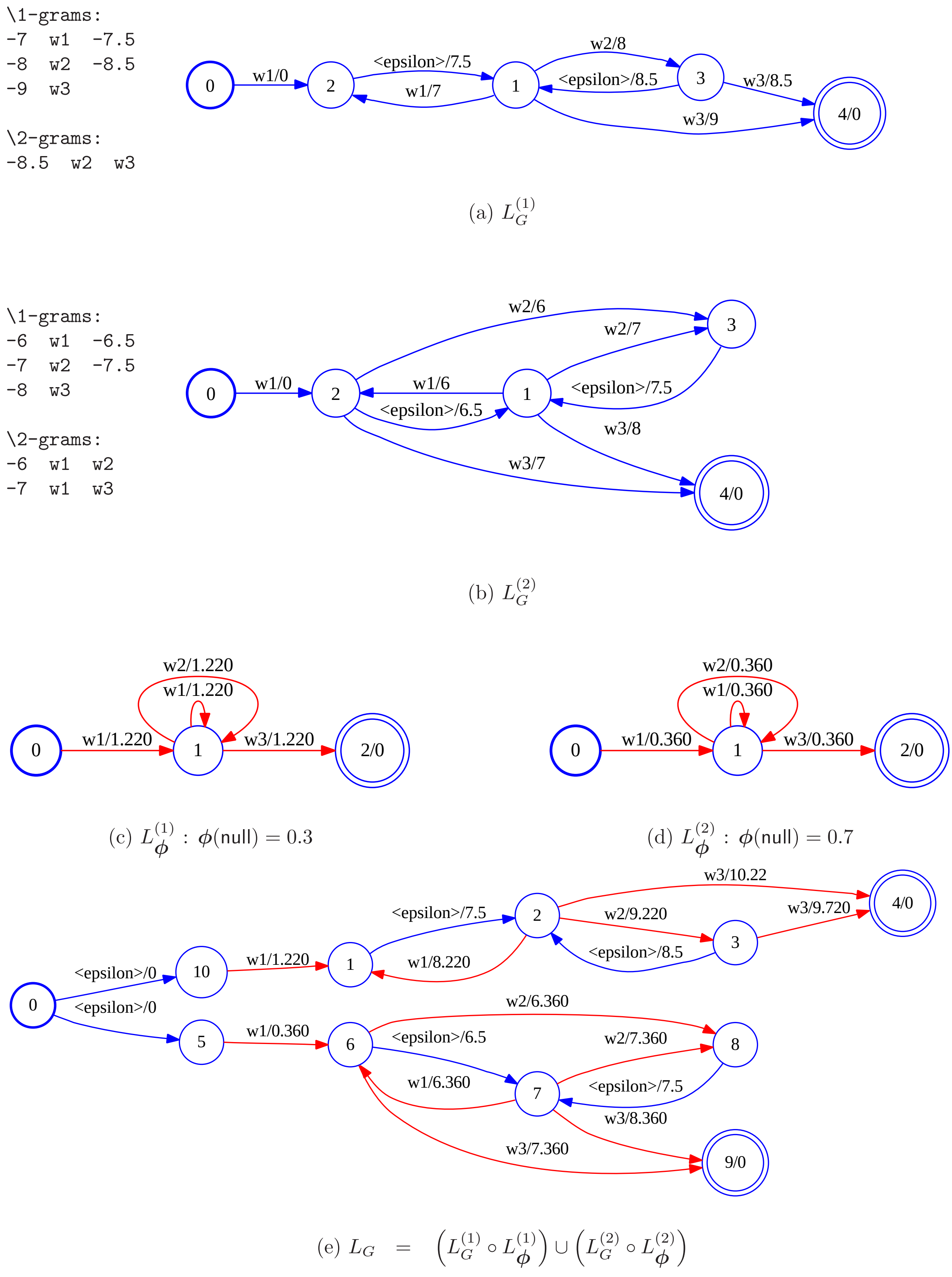
In contrast when using context dependent interpolation, more flexibility is introduced to the assignment of component weights as the history varies. For the two bi-gram component models of figure 2, their corresponding context dependent interpolation weight parameters and transducer representation are shown in figure 3(a) and 3(b). For bi-gram $P(w_3|w_2)$, a duly higher weight of 0.8 is now assigned to the first component model. The final interpolated LM's transducer representation is shown in figure 3(c). As expected, due to the nature of WFST composition operation, one noticeable effect is the expansion of the final transducer size. More distinct paths are now carrying different costs, as is manifested in the increased number of nodes and arcs in figure 3(c) compared with the context free interpolation of figure 2(e).

3.2 Continuous History Weighting Functions

One possible solution to the "curse of dimensions" associated with discrete context dependent weights is to use a continuous form of history weighting function. The basic approach is to convert discrete word history contexts to a continuous representation using a mapping function such as a neural network [1], or higher order polynomial interpolation [30]. Since the resulting component weight probabilities are smooth functions of the context representation, better generalization to unseen events may be achieved. The use of neural networks for continuous space language modeling has been studied for speech recognition in recent years [32]. The same concept may also be used to design a continuous history weighting function. An example of neural network based history weighting function is shown below.



Architecture of a neural network based history weighting function



```
\1-grams:
-7   w1   -7.5
-8   w2   -8.5
-9   w3

\2-grams:
-8.5   w2   w3
```

(a) $L_G^{(1)}$

```
\1-grams:
-6   w1   -6.5
-7   w2   -7.5
-8   w3

\2-grams:
-6   w1   w2
-7   w1   w3
```

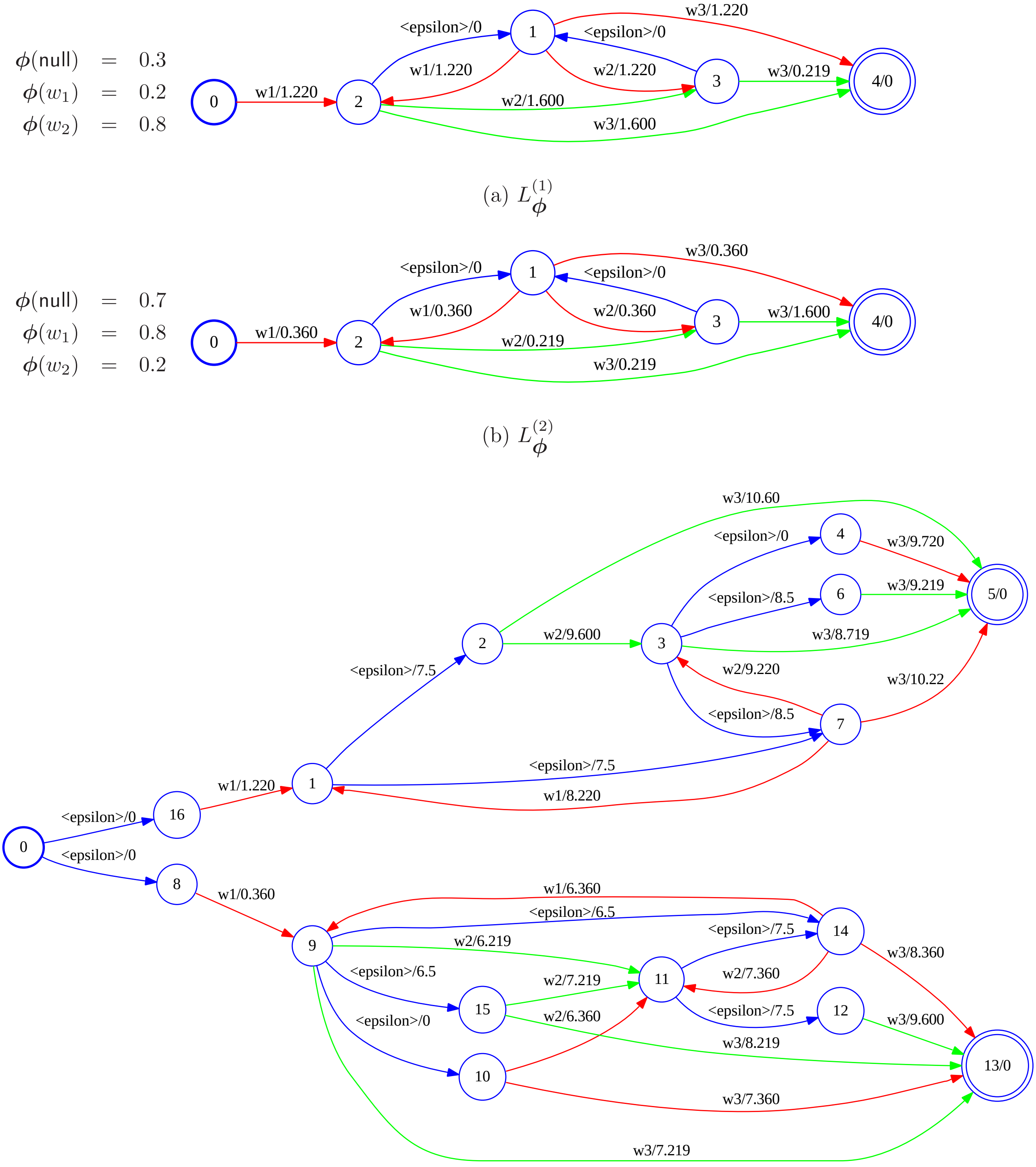
(b) $L_G^{(2)}$

(c) $L_{\phi}^{(1)}$: $\phi(\text{null}) = 0.3$

(d) $L_{\phi}^{(2)}$: $\phi(\text{null}) = 0.7$

(e) $L_G \;=\; \left(L_G^{(1)} \circ L_{\phi}^{(1)}\right) \cup \left(L_G^{(2)} \circ L_{\phi}^{(2)}\right)$

Figure 2: Two simple 2-gram back-off component language models and their WFST representation for a three word vocabulary $\{w_1, w_2, w_3\}$ in (a) and (b); the WFST representation of the context free interpolation weights are shown in (c) and (d); the WFST representation of the final linearly interpolated model is shown in (e). Arcs carrying the interpolation weight information are marked as red in all transducers.



(a) $L_{\boldsymbol{\phi}}^{(1)}$

(b) $L_{\boldsymbol{\phi}}^{(2)}$

(c) $L_G \quad = \quad \left(L_G^{(1)} \circ L_{\boldsymbol{\phi}}^{(1)}\right) \cup \left(L_G^{(2)} \circ L_{\boldsymbol{\phi}}^{(2)}\right)$

Figure 3: Two context dependent weight back-off models for the two component 2-gram models in figure 2(a), 2(b) and their WFST representation are shown in (a) and (b); the WFST representation of the linearly interpolated model using context dependent weights in (a) and (b) is shown in (c). Arcs carrying information from context free and dependent interpolation weights are marked as red and green respectively in all transducers.

Although continuous history weighting functions benefit from inherent smoothing and good generalization to unseen contexts, the weights computation may be slow. The rest of this paper focuses on using discrete context dependent weights for language model interpolation and adaptation.

4 Robust Estimation of Context Dependent Weights

As discussed, as the history length grows, the number of context dependent interpolation weights to estimate increases exponentially. During model tuning or test time adaptation, often only limited amount of data is available. This data sparsity issue is particularly important and must be addressed.

4.1 Model Interpolation Using Training Data

If a sufficiently large held-out data set is unavailable for estimating context dependent interpolation weights, the training data of component n-gram models may be used to ensure robust weights estimation. For discriminative training, this would require all data sources to have confusion explicitly generated. This is a non-trivial problem for sources of genres other than audio transcription. In this paper, the use of training data for a general model interpolation is only considered using ML.

When using training data of multiple text sources, the sufficient statistics derived from equations (2) and (14) in re-estimation will be dominated by large sized corpora, and thus introduce a bias. This is a fundamental issue that can affect performance of the interpolated model. Hence, such bias to corpus size must be addressed. The approach proposed in this paper is to use a corpus length normalization scheme. The training data log-probability given in equation (14) is modified as,

$$\ln P_{\mathsf{norm}}(\mathcal{W}) \quad = \quad \sum_m \frac{L}{L_m} \sum_{i=1}^{L_m} \ln P(w_i|h_i^{n-1}) \tag{18}$$

where L_m is the total number of words in the m^{th} corpus. The **normalized perplexity** (nPP) measure is computed using the above as

$$\mathcal{F}_{\mathsf{nPP}} \quad = \quad \exp\left\{-\frac{\ln P_{\mathsf{norm}}(\mathcal{W})}{\sum_m L}\right\} \tag{19}$$

Using the nPP criterion, weights are determined by the *average* word probability from each data corpus. As dependency upon word counts is removed, the bias to larger sized corpora can thus be handled.

As discussed in section 1, the variability among data sources and their contribution are jointly determined by a combination of multiple attributes. Some of them may be sufficiently modeled using global, context independent weights, for example, epoch and genre. Others such as modeling resolution, topics and styles, require local, context dependent weighting. Using the nPP criterion, interpolation weights at both levels can be estimated. In practice the global level diversity among sources is often further enlarged by conscious decisions when building component models. If certain sources are known to be useful for the domain of interest, for example, acoustic transcriptions, a bias to components of the same genre may be introduced during LM construction. When low cut-offs are used for these sources, the associated component models will have high probabilities on their training data compared to others built with more punitive cut-off settings. Similarly if robust discounting schemes are used then these models will also generalize well on other data. Using the nPP criterion, general LM interpolation using both context free and dependent weights may be robustly estimated on the training data. They can be used as standard LMs for decoding prior to test-time domain adaptation.

4.2 MAP Estimation

One common approach to address the robustness issue is to use maximum *a-posteriori* (MAP) estimation. Take the perplexity or ML based adaptation as an example, this is given by

$$\hat{\phi}_m(h_i^{n-1}) \quad = \quad \frac{\mathcal{C}_m^{\mathsf{ML}}(h_i^{n-1}) + \tau\phi_m^{\mathsf{Pr}}(h_i^{n-1})}{\sum_m \mathcal{C}_m^{\mathsf{ML}}(h_i^{n-1}) + \tau} \tag{20}$$

where $\mathcal{C}_m^{\mathsf{ML}}(h_i^{n-1})$ is context dependent ML statistics for history context h_i^{n-1}, and τ controls the contribution from weight prior, $\phi_m^{\mathsf{Pr}}(h_i^{n-1})$.

One key issue with MAP estimation is the choice of smoothing prior. In previous research the global, context free weights, $\phi(\mathsf{null})$, was used [21]. In order to introduce more context information, rather than completely backing off to the context independent weights, a hierarchical smoothing using weights of lower order contexts may also be considered, as inspired by interpolated Kneser-Ney smoothing of n-gram models [5]. Take the perplexity based estimation as an example, this is given by

$$\hat{\phi}_m(h_i^{n-1}) \quad = \quad \frac{\mathcal{C}_m^{\mathsf{ML}}(h_i^{n-1}) + \tau\hat{\phi}_m(h_i^{n-2})}{\sum_m \mathcal{C}_m^{\mathsf{ML}}(h_i^{n-1}) + \tau} \tag{21}$$

The form of hierarchical smoothing in equation (21) can also be applied to nPP and MBR based statistics.

When adapting LMs using context dependent interpolation, two sets of weights may be available. These are obtained from the training data nPP estimation and test set adaptation respectively. The first provides domain neutral and finer context dependent modeling, whilst the second give in-domain weights but potentially reduced context resolution. How to appropriately combine these two sources of information in order to handle this trade-off will be addressed in the later sections.

4.3 Class-based Approach

Class-based n-gram models have been shown to be helpful in addressing the data sparsity problem [2]. Words are clustered into syntactically, semantically or statistically equivalent classes. The intuition is that even if a word n-gram does not occur in the training data, the corresponding class n-gram may be available. To more robustly handle unseen or rarely observed events, a class based approach may also be used for context dependent weight estimation. Due to the high dimensionality of the history space, this is generally non-trivial for longer range contexts. The approach considered here is to perform the clustering at the word level. Interpolation weights are then shared among word histories that can be mapped to the same sequence of classes. This is given by,

$$P(w_i|h_i^{n-1}) \quad = \quad \sum_m \phi_m(\mathcal{G}_i^{n-1}) P_m(w_i|h_i^{n-1}) \tag{22}$$

where $\mathcal{G}_i^{n-1}$ is the preceding $n-1$ class history determined by a unique word to class mapping.

One key issue is how to derive a suitable word to class mapping. An efficient clustering scheme, referred to as *exchange algorithm*, has been proposed and widely used for standard class based n-gram models [16]. However, this algorithm may not be appropriate for context dependent weights, because class based n-gram models bear a fundamentally different form from the context dependent interpolation in equation (22). Therefore, an alternative clustering algorithms is required. The method considered is a maximum likelihood based *weight merging* algorithm explicitly derived for context dependent weights.

```
initialize each word in vocabulary as a distinct class;
for each word as history train 1-gram weight using ML;
iterate until the target number of classes obtained:
  find the pair of word classes whose merging gives the
  maximum likelihood gain or minimum loss;
  merge the found class pair into one class.
```

As discussed earlier, in order to reduce the bias to larger corpora in the clustering data, the normalized log-likelihood of equation (18) will be used. Note that directly computing the likelihood using equation (22) is infeasible, due to the iterative nature of weight estimation and the memory requirement to store component n-gram probabilities for all words in the clustering data. To handle this problem, the log-likelihood lower bound derived using Jensen's inequality is used instead:

$$\ln P_{\mathsf{norm}}(\mathcal{W}) \quad \geq \quad \sum_m \frac{L}{L_m} \sum_{i,m} \phi_m(g(w_{i-1})) \ln P_m(w_i|h_i^{n-1}) \tag{23}$$

Thus the sufficient statistics for likelihood computation simplify to the sum of component n-gram log probabilities for each word, w, that have itself as the immediate proceeding history, $\left\{\sum_{i,h_i^1=w} \ln P_m(w_i|h_i^{n-1})\right\}$. The change of log-likelihood bound only depends on the current pair of classes being merged, while all other classes are fixed. The weight estimates after a merging step can be derived from combining the ML weight statistics of the two classes before the merge, as given in (7).

In practice, the proposed weight merging scheme is found to consistently outperform the exchange algorithm for standard class based LMs. For example, table 1 shows performance of a range of interpolated language models using class context dependent weights with varying numbers of clusters on a 10 source, 1.0 billion word Mandarin Chinese broadcast news transcription task. In all configurations, the weight merging algorithm outperformed the baseline exchange algorithm on both training data nPP and test data PP scores.

Context Type	Clustering Algorithm	Num Class	nPP Trn	PP Test
1-gram class	exchange algorithm	50	76.1	217.9
		100	74.8	216.6
		200	73.5	214.8
		400	73.0	213.9
	weight merge	50	73.6	213.5
		100	73.4	213.1
		200	73.1	213.0
		400	72.8	212.7

Table 1: PP and nPP scores of interpolated language models using nPP trained 1-gram history dependent weights on a 10 source, 1.0 billion word Mandarin Chinese broadcast news transcription task.

5 Weight Set Combination

As discussed in section 4.2, when adapting LMs using context dependent interpolation, two sets of weights are available. These are obtained from the training data nPP estimation and test test adaptation respectively:

- ***training*** data nPP weights estimated using a hierarchical smoothing. They provide richer context information and finer modeling resolution, but potentially larger mismatch against the target domain during LM adaptation.
- ***test*** data self-adapted weights using equation (21) and a hierarchical smoothing. These provide a closer match to the target domain of interest. However, as the supervision may contain errors and not all contexts in the reference can have their own weights, a back-off to a lower order context based weights using equation (15) is necessary. This will result in reduced modeling resolution.

The above two sets of weight information provide either domain neutral, longer contexts based weights, or in-domain, shorter contexts based ones. In order to balance this trade-off, it is preferable to appropriately combine the two for context dependent LM adaptation. In the previous research largely relied on test set information. The combined use of the above two was limited. In this section four weight combination schemes are proposed to incorporate both training and test set information. They can be categorized into two broad types of techniques: two-stage MAP estimation and log-linear weight combination. Within each category, it is also optional to further supplement the adapted weights of contexts obtained from the training data with weights of contexts uniquely observed in the test set supervision. These contexts may carry additional information of the target domain for adaptation, and it is thus interesting to include them via a union operation. Note that the hierarchical smoothing of equation (21) effectively uses a lower order context based weight prior. However, for clarity in the rest of this paper the term “prior” is reserved and exclusively refers to nPP weights estimated on the training data.

A. Two-stage MAP Estimation: In the first stage nPP based LM interpolation is performed. Contexts are extracted from the training data. To improve robustness, their weights are MAP estimated using a hierarchical smoothing as in equation (21). In the second stage, test-time LM self-adaptation is performed, where the nPP estimated context dependent weights are used as a prior. For example, for ML based adaptation, the final adapted m^{th} component weight of history context h_i^{n-1} is given by

$$\hat{\boldsymbol{\phi}}_m^{\mathsf{comb}}(h_i^{n-1}) = \frac{\mathcal{C}_m^{\mathsf{ML}}(h_i^{n-1}) + \tau \hat{\boldsymbol{\phi}}_m^{\mathsf{nPP}}(h_i^{n-1})}{\sum_m \mathcal{C}_m^{\mathsf{ML}}(h_i^{n-1}) + \tau} \tag{24}$$

Note that in both stages all contexts are exclusively obtained from the training data.

B. Two-stage MAP Estimation and Union: This is an extension of option **A**. As discussed, contexts uniquely observed in the test set supervision may carry additional useful information of the target domain, their associated weights are also MAP adapted to the supervision using equation (21) and merged into the final combined context dependent weight set, rather than being discarded. The MAP adapted training data weight tree of figure 1 is effectively expanded by adding more nodes that represent the newly observed histories in the supervision. Alternatively using a WFST representation, the final combined transducer may be derived using

$$L_{\boldsymbol{\phi}_m}^{\mathsf{comb}} = L_{\boldsymbol{\phi}_m}^{\mathsf{MAP}} \cup \bar{L}_{\boldsymbol{\phi}_m}^{\mathsf{Supv}} \tag{25}$$

where $L_{\boldsymbol{\phi}_m}^{\mathsf{MAP}}$ is the transducer of the nPP weights MAP adapted to the test set supervision using equation (24), and $\bar{L}_{\boldsymbol{\phi}_m}^{\mathsf{Supv}}$ the weight transducer of contexts only observed in the adaptation supervision.

C. Log-linear Composition: MAP estimation may be viewed as a weighted linear interpolation, for example, between an ML estimate and its smoothing prior. There are two issues with this approach. First, training set contexts that are unavailable in the supervision will back-off to a domain neutral nPP prior containing minimum information of the test data. Second, due to the nature of linear interpolation, test set weights that are MAP adapted to incorrect supervision using option **A** can retain certain mis-ranking of component LMs. To address these issues, an alternative is to use a log-linear combination via WFST composition between the two interpolation weight sets. This is given by,

$$L_{\boldsymbol{\phi}_m}^{\mathsf{comb}} = L_{\boldsymbol{\phi}_m}^{\mathsf{nPP}} \circ L_{\boldsymbol{\phi}_m}^{\mathsf{Supv}} \tag{26}$$

where $L_{\boldsymbol{\phi}_m}^{\mathsf{nPP}}$ and $L_{\boldsymbol{\phi}_m}^{\mathsf{Supv}}$ are the two transducers that represent the nPP prior estimated on the training data and the context dependent weights adapted to the test data supervision. During this process, the longest matching contexts from each will be automatically found and used via the back-off process given in equation (15), rather than using potentially zero test set information for unseen events as in equation (24). Furthermore, a log-linear interpolation via WFST composition can also reject component LM weighting obtained from an erroneous supervision that are very different from the training set nPP prior. Therefore it can improve robustness of weight combination. In the first step, in common with option **A**, contexts are extracted from the training data and their associated nPP weights MAP estimated using a hierarchical smoothing. In the second step, contexts are extracted from the test set supervision and their weights MAP estimated using a hierarchical smoothing of equation (21). After a normalization to satisfy the sum-to-one constraint, the final combined weights are,

$$\hat{\boldsymbol{\phi}}_m^{\mathsf{comb}}(h_i^{n-1}) = \frac{\hat{\boldsymbol{\phi}}_m^{\mathsf{nPP}}(h_i^{n-1})^{\alpha} \hat{\boldsymbol{\phi}}_m^{\mathsf{bo}}(h_i^{n-1})}{\sum_m \hat{\boldsymbol{\phi}}_m^{\mathsf{nPP}}(h_i^{n-1})^{\alpha} \hat{\boldsymbol{\phi}}_m^{\mathsf{bo}}(h_i^{n-1})} \tag{27}$$

where α is a tunable log-linear scaling factor that controls the contribution from the nPP prior. In this option it is un-tuned and set as $\alpha = 1.0$. In the same fashion as **A**, new contexts unique to the test set supervision are discarded.

D. Weighted Log-linear Composition and Union: This is a modified form of option **C**. For any context extracted from the training data, if it has no matching context of any length at all in the test set supervision and therefore completely backs off to the context free, global weights, equation (27) ($\alpha = 1.0$) is still used to obtained the combined weights. Otherwise, the test data supervision adapted weights for the longest matching context will be used. This is effectively achieved by setting $\alpha = 0$ in equation (27). In common with **B**, weights of contexts uniquely observed in the test set supervision are also added. Using a WFST representation, this is given by,

$$L_{\boldsymbol{\phi}_m}^{\mathsf{comb}} = \left(\bar{L}_{\boldsymbol{\phi}_m}^{\mathsf{nPP}} \circ L_{\boldsymbol{\phi}_m}^{\mathsf{Supv}} \right) \cup L_{\boldsymbol{\phi}_m}^{\mathsf{Supv}} \tag{28}$$

where $\bar{L}_{\boldsymbol{\phi}_m}^{\mathsf{nPP}}$ the weight transducer representing only contexts that are observed in the training data but not those in the adaptation supervision. Compared with **C**, this approach is leaning more to the estimates from the test set supervision whenever context dependent weights are available. Hence, it is closer to the target domain for LM adaptation.

6 Implementation Issues

In this section a number of implementation issues that may affect performance of context dependently language models interpolation and adaptation are discussed.

Computation Efficiency: The use of context dependent weights dramatically increase the computational cost when building interpolated language models. During estimation and decoding, efficient access is required when applying context associated weights to calculate the interpolated probabilities. In this work, these context dependent interpolation weights are stored in a tree structure parallel to the component n-gram LMs. Let V denote the vocabulary size. The access cost to any context through the tree is $\mathcal{O}((n-1) \times V)$. The use of a tree structure is particularly useful for the MAP estimation with a lower order context based prior discussed in section 4.2 and the weight back-off of section 3. To further improve efficiency, hash tables are used to cache weights of the most frequent contexts.

Weights Initialization: As discussed in section 2, global, context independent weights are commonly tuned on some held-out data when interpolating source specific LMs. During test-time LM adaptation, it is possible to use these tuned weights as initialization. However, when using the training data and nPP criterion for LM interpolation, no test data knowledge is required. For consistency reasons, equal weights based initialization are used for all experiments conducted in this paper, unless otherwise stated.

Use of N-best Lists: Multiple hypotheses are required to accumulate the sufficient statistics given in equation (12) for MBR estimation. In this paper, the top N-best 1000 hypotheses are generated using the baseline language model interpolated with context free, global weights tuned on held-out data. These N-best lists are kept fixed for all experiments. In order to further reduce memory requirement, the word probabilities required by the statistics given in equation (12) for MBR estimation are generated off-line for each N-best candidate using each component LM and kept fixed.

Cost Function: MBR estimation requires the calculation of a error cost function. For the Mandarin Chinese broadcast transcription task considered in this paper, the cost function used is the character error rate (CER). The 1-best output from the unadapted baseline system will be used to compute the CER scores over all the N-best entries for every speech segment. For supervised adaptation, reference transcriptions based on manual segmentation are first mapped onto automatically derived speech segment boundaries prior to CER computation.

MAP Smoothing Constant τ: As discussed in section 4.2, the constant τ controls the contribution from a weight prior during MAP estimation. Its appropriate setting depends on the nature of the statistics used in MAP estimation. When using the nPP criterion, inverse corpus size weighted, normalized statistics are accumulated to estimate interpolation weights. The appropriate setting of τ can therefore be different from those used in standard perplexity or MBR based test-time model adaptation. In this paper, it is set as $\tau = 2.5$ and fixed for all perplexity and MBR adaptation experiments, and $\tau = 100$ for nPP LM interpolation using the training data.

EBW Smoothing Constant D: As discussed in section 2.2, the setting of the smoothing constant, D, may affect both the optimization stability and generalization of MBR estimation. As in discriminative training of HMMs, its setting is largely chosen on heuristics and empirical basis [29]. The form considered in this paper is $D = E \times N_{\mathcal{W}}$, where $N_{\mathcal{W}}$ is the number of speech segments in the supervision data, and $E > 0$, typically set as 50. In practice this was found a good compromise between convergence speed and generalization. Varying E was also found having minimum effect on performance. In this paper E is always set as 50 and never altered.

Decoding with context dependent LM interpolation: When using context dependent interpolation weights in decoding, there is a flexible choice between a static, off-line application, and dynamic, on-the-fly application of the weights. For language models interpolated using the nPP criterion, an off-line interpolation is preferred. For test-time adaptation, it is more efficient to use a dynamic application of component n-gram probabilities and interpolation weights for each context. As discussed in section 3.1, the composition between component n-gram and their weight models lead to the expansion of the final transducer size. Hence, compared with using a global, context free interpolation, there can be more paths with unique LM scores that need to be kept distinct. This is particularly true for class based context dependent weights. In practice, such effect can lead to a significant lattice size increase from 20% to 120% over baseline lattices expanded using LMs with context free interpolation.

7 Experiments and Results

In this section experimental results on a Mandarin Chinese broadcast speech transcription task are presented. First, description of the baseline LVCSR system is given. Then performance of various interpolated and adapted languages models are evaluated. Finally, experimental results on using integrated weight estimation in model interpolation and adaptation are presented.

7.1 Baseline System Description

The CU-HTK Mandarin ASR system was used to evaluate LMs using various interpolation and adaptation techniques [33]. It comprises an initial lattice generation stage using a 58k word list, interpolated 4-gram word based back-off LM, and adapted MPE acoustic models trained on 942 hours of broadcast speech data. A total of 1.0G words from 10 text sources were used in LM training. Information on corpus size, cut-off settings and smoothing schemes for component LMs are given in table 2. For data sources that are closer in genre to the test data, minimum cut-offs and modified KN smoothing were used. These include the two acoustic transcriptions sources, bcm and bnm, and additional data collected from major TV channels or Chinese media such as CCTV, VOA and Phoenix TV . For the two largest corpora of newswire genre, giga-xin and giga-cna, more aggressive cut-offs and Good Turing (GT) discounting were used. Again, these conscious decisions are often made in state-of-the-art LVCSR systems when certain text sources are known to be more useful for the target domain, as discussed in section 4.1. Three Mandarin broadcast speech evaluation sets were used: bn06 of 3.4 hour BN data, bc05 of 2.5 hours of BC data and the 1.8 hour GALE 2006 evaluation set eval06.

Comp LM	Text (M)	Train Config	Global Weight Tuning PPTest	 PPTrn	 nPPTrn
bcm	4.83	111,kn	0.2325	0.0049	0.1426
bnm	3.78	111,kn	0.1327	0.0066	0.1729
giga-xin	277.6	123,gt	0.1389	0.2428	0.1079
giga-cna	496.7	123,gt	0.1734	0.4577	0.0815
phoenix	76.89	112,kn	0.1030	0.1125	0.1225
voarfabbc	30.28	112,kn	0.1026	0.0270	0.0734
cctvcnr	26.81	112,kn	0.0404	0.0391	0.0844
tdt4	1.76	112,kn	0.0250	0.0060	0.0717
papersjing	83.73	122,kn	0.0285	0.0919	0.0928
ntdtv	12.49	122,kn	0.0324	0.0114	0.0503

Table 2: Text source, 2/3/4-gram cut-off settings, smoothing scheme used in training and global ML weights tuned using test set PP (bn06+bc05), training data PP and nPP scores for component language models.

7.2 Performance of Interpolated Language Models

PP and nPP scores for various interpolated LMs are presented in table 3. The first two lines of the table show the performance of using equal, or global, context free weights tuned on the perplexity of combined bn06+bc05. The latter case is the standard form of model interpolation for current ASR systems. These weights are in the fourth column of table 2. The third line shows the performance of weights tuned using the training data PP metric. Compared with the equal weighted interpolation, there was a significant degradation of 24 to 150 PP points on the all test sets. Similarly there is a large error rate increase of 0.3%-1.4% absolute. This is due to the corpus size bias discussed in section 4.1. Such a bias further manifests itself in the global weights given in the 5th column of table 2. The largest two corpora, giga-cna (0.46) and giga-xin (0.24) were heavily weighted.

Using the nPP metric in weight estimation, this bias was greatly reduced, as given in the fourth line of table 3. The corresponding global weights are in the 6th column of table 2. Large sized corpora no longer dominate the weight assignment. As discussed in sections 1 and 4.1, the weights are determined by a combination of global factors, such as source of collection, epoch and genre, and local factors including modeling resolution, generalization, topics and styles. During the nPP estimation, if a particular component model is both underfitting to its own training data and generalizing poorly to other sources, its weight is likely to be low. For example, the biggest newswire source giga-cna, of Taiwanese origin and different in style from other broadcaster sources, trained using aggressive cut-offs and simple GT discounting, is now weighted by 0.082. In contrast, the acoustic transcription source bnm, collected from major mainland Chinese broadcasters, similar in genre, topics and style to most of other data sources in the table, and trained with minimum cut-offs and more robust KN smoothing, is weighted by 0.17. It is also interesting to note that if minimum cut-offs and KN smoothing were used to build all source specific models and no conscious bias is introduced, a much smoother weight distribution can be obtained. For example, the weight assigned to bnm is decreased to 0.13. This is expected as improved modeling resolution and smoothing scheme would help previously under-weighted sources to be more useful during nPP estimation, and therefore on a more competitive footing against other sources. Using "nPP" system with context free LM interpolation weights , consistent PP improvements were obtained on all test sets against the equal weight baseline. In particular, PP reductions more than 10 points (4.6%-4.8% rel) were obtained on bc05 and eval06. Using this language model, CER performance comparable to the supervised baseline "PP.base" is also obtained.

Train Crit	Wgt Cntxt Type/Len	Train nPP	Reference Perplexity			CER%		
			bn06	bc05	eval06	bn06	bc05	eval06
eql	-/-	84.1	200.9	251.9	246.5	8.4	19.3	19.1
PP.base	-/-	84.0	194.6	227.1	231.7	8.4	19.0	19.1
PP.trn	-/-	130.8	224.5	403.8	360.7	8.6	20.7	19.9
nPP	-/-	82.0	197.6	239.9	235.3	8.3	19.1	19.1
nPP	word/1g	69.7	193.3	224.1	221.9	8.1	19.1	19.1
	word/3g	51.9	179.3	213.1	215.2	8.1	19.0	18.7
	cls100/1g	73.7	196.1	228.5	225.6	8.2	19.0	19.0
	cls100/3g	67.1	192.3	220.9	221.2	8.2	19.1	19.0

Table 3: PP and lattice rescoring 1-best CER% performance of interpolated LMs on bn06, bc05 and eval06. Global interpolation weights of "PP.base" baseline tuned on the reference of combined bn06+bc05 set. Equal weights initialization for all models.

These results suggest the nPP criterion may be used as an alternative LM interpolation technique. For robust estimation of context dependent interpolation weights on the training data, the nPP based approach becomes even more useful. Performance of two word based and two class based systems are shown in the last four lines of table 3. The two class based models were built using the 100 classes derived using the weight merging algorithm of section 4.3 and given in the bottom section of table 1. Due to memory constraint in weight estimation, the three word history based nPP model was built by extracting word level contexts from the "word/1g" nPP model after being pruned at 1.0e-9. The more compact class based contexts were extracted from the same model without pruning. The total number of interpolation weight vectors for word or class based histories of varying length and hit rate statistics for non-OVV tokens in the reference transcription of bn06, bc05 and eval06 are summarized in table 4. It is interesting to notice that for the word based system two word or three word context dependent weights provide a good coverage of more than 80% of the contexts found in the reference transcriptions. As expected, the class based weights have an even lower percentage of back-off to lower order contexts. Across all three test sets, three 3-gram class history based interpolation weights are predominantly used at a rate of more than 85%.

Context Type	#History Context			Reference Hit Rate%(1g/2g/3g)		
	1g	2g	3g	bn06	bc05	eval06
word	58k	5.2M	1.9M	16.9/55.4/27.7	16.1/55.2/28.8	19.2/53.5/27.3
cls100	100	7.2k	118k	3.78/4.08/92.1	5.75/5.66/88.6	6.71/6.91/86.4

Table 4: The number of interpolation weight vectors for word or 100 class history based contexts of varying length in nPP estimation and their hit rates for non-OVV tokens in the reference of bn06, bc05 and eval06.

Now it is interesting to investigate whether the above hit rates for various forms of context dependent interpolation weights can be transformed into improved generalization and error rate performance on the test data. These were trained using the nPP estimation with the hierarchical weight smoothing given in equation (21). Using 1-gram word level weights, there are 10 points of PP improvement on eval06, and an absolute CER reduction of 0.3% over the "PP.base" baseline on bn06. Increasing the context span to 3-gram word based history gave the best PP and CER performance. Compared with the equal weight baseline, 22 to 36 points of PP improvements (10%-17% rel) were obtained over all test sets The corresponding CER reductions are 0.2%-0.4% absolute. This 3-gram word context based nPP system also outperformed the supervised baseline “PP.base” on both perplexity and error rate. For example, statistically significant CER improvements of 0.3% and 0.4% absolute were obtained on bn06 and eval06 respectively. The last two lines of table 3 shows the performance of two class based systems. The two class based systems gave smaller CER improvements of 0.1%-0.2% absolute over the equal weight baseline. These trends appear to contradict the high hit rates of three word history based weights shown in table 4, and suggest a possible over-generalization of class based interpolation weights. The likelihood of many other hypotheses that have sub-optimal error rates but correspond to the same class sequence may be unduly boosted. Hence, additional confusion may be introduced during search. Overall, there is a consistent and strong correlation between training data nPP and test set PP scores in table 3.

7.3 Performance of Adapted Language Models

Now it’s interesting to examine the performance of adapted LMs using context dependent interpolation weights. LM adaption was performed at the audio show level. Equal weight initialization was used. The smoothing constant setting $\tau = 2.5$ were used in the form of MAP adaptation given in equation (21). The MBR smoothing constant D is set as in section 6. A total of 8 iterations of weights re-estimation were performed. The 4-gram lattice 1-best output generated by the tuned baseline with global weights (second line of table 3) was used as the adaptation supervision. The number of single word, two word and three word history contexts extracted from the supervision for the three test sets and the associated hit rate statistics for non-OVV tokens in the reference transcriptions are shown in the table 5. Again, the 100 classes derived using the weight merging algorithm given in the bottom section of table 3 were used for class based context dependent weights. Consistent with the hit rate statistics for nPP context dependent interpolation shown in table 4, class based weights have considerably higher 3-gram history weight hit rates than word based ones. The top 1000 hypotheses were extracted for MBR adaptation. During search component language models were interpolated on the fly using adapted interpolation weights for lattice rescoring.

Context Type	#History Context			Reference Hit Rate%(1g/2g/3g)		
	1g	2g	3g	bn06	bc05	eval06
word	32k	70k	78k	15.7/10.0/69.6	26.7/19.4/49.8	25.4/13.3/53.7
cls100	2.1k	14k	38k	5.43/10.4/84.2	7.02/14.4/78.6	11.2/20.7/67.8

Table 5: The number of interpolation weight vectors for word or 100 class history based contexts of varying length extracted from the supervision on bn06, bc05 and eval06 for LM adaptation, and their hit rate statistics for non-OVV tokens in the reference transcription.

As discussed in section 4, an important issue during MAP estimation of context dependent weights is the form of the smoothing prior. Experiments in this section only consider using a test data based prior estimated during adaptation. The use of a static nPP prior estimated on the training data as proposed in section 5 will be evaluated in later sections. It is found that the hierarchical weight smoothing given in equation (21), which uses a more informative lower order parental context based prior, consistently outperforms a global, context free one, as is shown in table 6. Hence, a test data based hierarchical weight smoothing is used for all experiments presented in rest of this section.

Wgt Prior	Wgt Cntxt Type/Len	Reference Perplexity			CER%		
		bn06	bc05	eval06	bn06	bc05	eval06
glob	word/3g	138.2	188.1	187.7	8.1	18.9	18.8
	cls100/3g	140.7	191.0	189.5	8.0	18.9	18.7
hier	word/3g	132.9	176.0	184.2	8.0	18.8	18.7
	cls100/3g	128.0	179.6	171.6	8.1	19.0	18.9

Table 6: CER and PP performance of ML adapted 4-gram LMs on bn06, bc5 and eval06, using global, context free or lower order history based hierarchical prior for smoothing. Equal weights initialization for all models.

Performance of ML adaptation are shown in table 7. Using global, context free PP based adaptation, there are

26 to 45 points of PP improvements (12%-23% rel) for all sets over the unadapted baseline system in the first line of table 7 (also show as "PP.base" in the second line of table 3). Absolute CER gains of 0.3% on bn06 and eval06 were obtained. Using context dependent adaptation, a further PP reduction of 13 points (8% rel) was obtained by the word level 1-gram weights. However, the CER gains were marginal. Using longer 3-gram word context based weights gave the best adaptation performance. This gave a PP reduction of 18 to 25 points (8%-12% rel) over all test sets. Again, these only transformed into marginal CER gains of 0.1% absolute. A similar trend can also be found in the two more compact class based systems. For example, using the 3-gram class based system, there are 47 points of PP reduction (21% rel) on bc05 against the unadapted baseline, but no CER improvement was obtained. These results suggest a weak correlation between PP and error rate. ML based language model adaptation may improve PP on the common contexts observed in both the supervision and reference, but not necessarily helpful in generalization and discrimination. Hence, it would be interesting to evaluate the performance of MBR adaptation.

LM Adapt	Wgt Cntxt Type/Len	Reference Perplexity			CER%		
		bn06	bc05	eval06	bn06	bc05	eval06
-	-/-	194.6	227.1	231.7	8.4	19.0	19.1
glob	-/-	150.0	201.1	200.9	8.1	18.9	18.8
context	word/1g	138.5	188.3	187.7	8.1	18.9	18.7
	word/3g	132.9	176.0	184.2	8.0	18.8	18.7
context	cls100/1g	146.0	195.8	194.2	8.1	18.9	18.7
	cls100/3g	128.0	179.6	171.6	8.1	19.0	18.9

Table 7: CER and PP performance of ML adapted 4-gram LMs on bn06, bc5 and eval06. Equal weights initialization for all adapted models. Context dependent adaptation uses lower order context prior for smoothing.

These are shown in table 8. Using context free or 1-gram weights the MBR systems gave comparable performance to the ML baselines in table 8. Increasing the context length to 3 further reduced the CER. The best performance was obtained using 3-gram word level context based weights, which gave statistically significant CER reductions of 0.3% on bn06, 0.4% on bc05 and 0.5% on eval06 over the unadapted baseline. Consistent with the results in table 7, the two word based MBR systems outperformed the class based ones. In the reset of this paper, only word based context dependent LM adaptation is considered. As expected, a fairly strong correlation is also found between the MBR criterion (expected character error rate) and CER in table 8.

LM Adapt	Wgt Cntxt Type/Len	Reference MBR Crit			CER%		
		bn06	bc05	eval06	bn06	bc05	eval06
-	-/-	14.86	22.66	20.52	8.4	19.0	19.1
glob	-/-	14.68	22.56	20.32	8.1	18.9	18.8
context	word/1g	14.61	22.47	20.22	8.1	18.7	18.6
	word/3g	14.60	22.43	20.21	8.0	18.6	18.6
context	cls100/1g	14.67	22.56	20.32	8.0	18.9	18.7
	cls100/3g	14.66	22.56	20.31	8.1	18.8	18.7

Table 8: MBR criterion and CER performance of MBR adapted 4-gram LMs on bn06, bc5 and eval06. Equal weights initialization was used. Context dependent adaptation uses lower order context prior for smoothing.

7.4 Combined Use of Interpolation and Adaptation Weights

As discussed in section 5, when adapting LMs using context dependent interpolation, two sets of weights are available. These are obtained from the training data nPP estimation and test test adaptation respectively. The trade-off between using domain independent, longer context weights estimated on the training data, and in-domain, shorter context weights adapted in test-time is an important issue. In this section experiments are conducted to evaluate a range of weight combination methods proposed in section 5. PP and CER performance of various language models are shown in table 9. The first line is the baseline "PP.base" system of table 3 using context free interpolation. Performance of three ML adapted LMs without using training set information are also shown in the table from the second to fourth line. They were also previously given in the same lines of table 7 for systems using context free, single word, or three word context dependent LM adaptation respectively.

Context		Wgt	CER%/PP(Reference)		
Prior	Adapt	Com	bn06	bc05	eval06
-	Supv	-	8.4/194	19.0/227	19.1/232
	-	-	8.1/150	18.9/201	18.8/201
	1		8.1/139	18.9/188	18.7/188
	3g		8.0/133	18.8/176	18.7/184
3g	Supv	-	8.1/179	19.0/213	18.7/215
	3g	**A**	8.1/128	18.9/176	18.7/179
		B	8.0/120	18.9/168	18.7/171
		C	8.0/126	19.0/180	18.7/180
		D	7.9/118	18.8/166	18.5/169

Table 9: CER and PP performance of ML adapted 4-gram LMs on bn06, bc05 and eval06.

Performance of combining training set nPP and test adapted weights using the methods proposed in section 5 are shown in the last four lines of table 9. The output from the nPP system in table 3 was used as the adaptation supervision, as is shown in the 5th line of table 9. The two-stage MAP estimation (option **A**) and log-linear composition approaches (option **C**) gave similar performance. Using the a two-stage MAP estimation with union (option **B**), further PP reduction was obtained but the CER gains were minimum. The weighted log-linear composition and union approach (option **D**) gave the best performance among the four. More than 30 points of PP reduction (17%-22% rel) and 0.1%-0.3% absolute CER reduction were obtained over the adapted baseline using context free weights (2nd line of table 9). The associated hit rate statistics on the reference transcription for this system on various test sets are shown in the bottom line of table 10. The first line of the table shows the context hit rates of using the nPP prior model alone. This was also shown previously in the first line of table 4. Compared with the hit rates of using the weights obtained from test set adaptation alone shown in the second line, the combined weight set obtained using option **D** consistently improved the context coverage on the reference transcriptions for all three sets. These results suggest sufficient coverage of contexts in test set supervision is important when adapting LMs using context dependent interpolation.

Context	Reference Hit Rate%(1g/2g/3g)		
	bn06	bc05	eval06
train	16.9/55.4/27.7	16.1/55.2/28.8	19.2/53.5/27.3
test	15.7/10.0/69.6	26.7/19.4/49.8	25.4/13.3/53.7
comb	17.5/12.5/70.1	28.4/21.7/50.4	28.4/17.6/54.4

Table 10: The hit rate statistics for non-OOV tokens in the reference transcription of bn06, bc05 and eval06 for word level context dependent interpolation weight vectors of varying length. These were obtained from the nPP prior weight set, extracted and adapted to the test data supervision, or the final combined weight set derived using option D.

ML adaptation may improve PP on the common contexts observed in both the supervision and reference, but not necessarily helpful in generalization and discrimination. Hence, it is now interesting to investigate the performance of MBR discriminative adaptation. These are shown in table 11. Performance of three MBR adapted systems with context free or word context dependent weights are shown from the 2nd to 4th line These are also shown in the same lines of table 8. The 3-gram weight MBR system gave the best adaptation performance. Using a weighted log-linear composition and union based approach (option **D**) to combine with the nPP system of table 3 gave further CER improvement of 0.2% on bc05. The CER gains over the adapted baseline using context free interpolation weights are 0.2% on bn06 and 0.3% on bc05 and eval06. The total CER gains over the unadapted baseline system are 0.5% (6% rel) on bn06, 0.4% on bc05 and 0.6% on eval06, all being statistically significant.

Context		Wgt	CER%		
Prior	Adapt	Com	bn06	bc05	eval06
-	Supv	-	8.4	19.0	19.1
	-	-	8.1	18.9	18.8
	1g		8.1	18.7	18.6
	3g		8.0	18.6	18.6
3g	3g	**D**	7.9	18.6	18.5

Table 11: CER performance of MBR adapted 4-gram LMs on bn06, bc05 and eval06.

8 Conclusion

Context dependent form of language model interpolation and adaptation using a discriminative method was investigated in this paper. A novel LM interpolation technique using normalized perplexity was proposed to robustly estimate context dependent language model interpolation weights on the training data. MAP estimation of back-off weights and class based schemes were also used to address the data sparsity problem. Several forms of smoothing priors were proposed. An efficient bottom-up maximum likelihood clustering algorithm was derived for the class based approach. A range of schemes to integrate weight estimation in LM interpolation and adaptation were proposed. Experimental results on a state-of-the-art Mandarin broadcast speech transcription task suggest that context dependent language model interpolation and adaptation may be useful for speech recognition and other related tasks, such as statistical machine translation. Future research will focus on using discriminative training techniques in both model interpolation and adaptation stages. Continuous forms of history weighting function will also be investigated.

References

[1] Y. Bengio & R. Ducharme (2003). "A neural probabilistic language model," in *Advances in Neural Information Processing Systems*, vol. 13. 2001. Morgan Kaufmann.

[2] Brown, P.F., Della Pietra, V.J., deSouza, P. V., Lai, J.C. and Mercer, R.L. (1992). Class-based n-gram models of natural language. *Computational Linguistics* 18 (4), 467470.

[3] I. Bulyko, M. Ostendorf & A. Stolcke. "Getting More Mileage from Web Text Sources for Conversational Speech Language Modeling using Class-Dependent Mixtures", in *Proc. HLT'03.*

[4] I. Bulyko, S. Matsoukas, R. Schwartz, L. Nguyen & J. Makhoul (2007). Language Model Adaptation in Machine Translation from Speech, in *Proc. ICASSP'07*, Hawaii.

[5] Chen, S.F., Goodman and J.T. (1999). An empirical study of smoothing techniques for language modeling. Computer Speech and Language 13 (4), 359394.

[6] Z. Chen, M. Li & K.-F. Lee (2000). Discriminative Training of Language Model, in *Proc. ICSLP'00*, Beijing.

[7] J. Darroch & D. Ratcliff (1972). "Generalized iterative scaling for log-linear models", Ann. Math. Statist., vol. 43, 1972.

[8] V. Doumpiotis & W. Byrne. Lattice segmentation and minimum Bayes risk discriminative training for large vocabulary continuous speech recognition. In *Speech Communication*, (2):142-160, 2005.

[9] Emami, A., Jelinek, F. (2005). Random clusterings for language modeling. In *Proc. ICASSP'05*, Philadelphia.

[10] P. S. Gopalakrishnan, D. Kanevsky, A. Nádas, & D. Nahamoo (1991). An Inequality for Rational Functions with Applications to Some Statistical Estimation Problems, *IEEE Transactions on Information Theory*, January, 1991.

[11] G. Hinton. Training Products of Experts by Minimizing Contrastive Divergence, *Neural Computation*, 14:1771–1800, 2002.

[12] B. Hsu (2007), Generalized Linear Interpolation of language Models (2007). *Proc. IEEE ASRU'07*, Kyoto.

[13] F. Jelinek. *Statistical Methods for Speech Recognition.* MIT Press, Cambridge, Massachusetts, 1997.

[14] S. M. Katz (1987). Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 35 (3), 400401.

[15] J. Kaiser, B. Horvat & Z. Kacic, A Novel Loss Function for the Overall Risk-criterion Based Discriminative Training of HMM Models, *Proc. ICSLP'00*, Beijing.

[16] R. Kneser and H. Ney (1993), "Improved clustering techniques for class based statistical language modeling," in *Proc. of Eurospeech93'*, 1993.

[17] H-K. J. Kuo, E. Fosler-Lussier, H. Jiang & C-H. Lee (2002). Discriminative Training of Language Models for Speech Recognition, in *Proc. ICASSP'02*, Florida.

[18] H-K. J. Kuo & B. Kingsbury (2007). Discriminative Training of Decoding Graphs for Large Vocabulary Continuous Speech Recognition, in *Proc. ICASSP'07*, Hawaii.

[19] S.-S. Lin & F. Yvon (2005). Discriminative training of finite state decoding graphs, in *Proc Interspeech'05.*

[20] X. Liu, W. J. Byrne, M. J. F. Gales & P. C. Woodland et al. (2007). Discriminative Language Model Adaptation for Mandarin Broadcast Speech Transcription and Translation, in *Proc. IEEE ASRU'07*, Kyoto.

[21] X. Liu, M. J. F. Gales &P. C. Woodland (2008). Context Dependent Language Model Adaptation, in *Proc. Interspeech'08*, Brisbane.

[22] S. Martin, J. Liermann & H. Ney (1998). "Algorithms for bigram and trigram word clustering," *Speech Communications*, vol. 24, no. 1, pp. 19-37, April 1998.

[23] D. Mrva & P. C. Woodland (2006). Unsupervised Language Model Adaptation for Mandarin Broadcast Conversation Transcription, in *Proc. ICSLP'06*, Korea.

[24] M. Mohri. Finite-state transducers in language and speech processing. *Computational Linguistics*, 23:2, 1997.

[25] M. Mohri & M. Riley. Network optimizations for large vocabulary speech recognition. *Speech Communication*, 25:3, 1998.

[26] M. Mohri, M. Riley, D. Hindle, A. Ljolje & F. Pereira (1998). Full expansion of context-dependent networks in large vocabulary speech recognition. In *Proc. ICASSP'98*, Seattle, Washington, 1998.

[27] M. Mohri, F. Pereira & M. Riley (2000). Weighted Finite-state Transducers in Speech Recognition, in *Proc. ASR'00*, 2000.

[28] F. J. Och & H. Ney. Discriminative Training and Maximum Entropy Models for Statistical Machine Translation. In *Proc. ACL02'*, pp. 295-302, Philadelphia.

[29] D. Povey & P. C. Woodland (2002). Minimum Phone Error and I-smoothing for Improved Discriminative Training, *Proc. ICASSP'02*, Florida.

[30] W. Press et al. (1992). *Numerical Recipes in C. The Art of Scientific Computing (2nd edition)*, Cambridge University Press. ISBN 978-0-521-43108-8.

[31] B. Roark, M. Saraclar & M. Collins (2006). Discriminative n-gram language modeling, *Computer Speech and Language*, 2006.

[32] H. Schwenk (2007). Continuous SpaceLanguage Models, *Computer Speech and Language*, 21:492-518, 2007.

[33] R. Sinha, M. J. F. Gales, D. Y. Kim, X. Liu, K. C.Sim, and P. C. Woodland (2006). The CU-HTK Mandarin broadcast news transcription system, in *Proc. ICASSP'06.*

[34] A. Webb (1999) Statistical Pattern Recognition, Oxford University Press, 1999.