Mixtures of Word and Class Language Models using Context-Dependent Mixture Weights*

E. W. D. Whittaker (Tokyo Institute of Technology)

1 Introduction

It is well known that linearly interpolating a class model with a word model decreases the perplexity (PP) and typically also the word error rate (WER) of automatic speech recognition systems. We investigate robustly estimating more specific context-dependent interpolation weights for a word model and a large number of different class models. The combination method we propose is similar to deleted interpolation [2] where held-out data is used to estimate weights dependent on the frequency of occurrence of the context. Typically there is not enough held-out data to estimate more refined weights and only one global weight per component language model is estimated by maximizing the log-likelihood of the model combination on some (typically) small held-out set [3]. In this paper, we use the (typically) large training set to estimate the increased number of additional weights by maximizing the leaving-one-out (LOO) log-likelihood of the training data [4]. LOO is a special case of deleted estimation in which the text is partitioned into \( N_W \) parts, where \( N_W \) is the size of the corpus, so each held-out sample constitutes only one word, or event.

2 Class-based language models

In the literature, the most commonly used formulation of class model is the two-sided model:

\[
P(w_i | h) = P(w_i | C_i(w_i)) \cdot P(C_i(w_i) | C(h)) 
\]

where the position-dependent class mappings \( C_i \) are deterministic and are all be identical. This model has been shown to have desirable features such as fast clustering times and complimentary performance with other class models [5].

3 Context-dependent interpolation weights

Linear interpolation is often the technique of choice in language modelling for combining models to exploit complementary features of the component models. In the literature, with few exceptions, global interpolation weights \( \lambda_m \) are used i.e. one global weight for each of the \( M \) component models:

\[
P_{\text{interp}}(w | h) = \sum_{m=1}^{M} P_m(w_i | h) \cdot \lambda_m, \tag{4}
\]

where \( \sum_{m=1}^{M} \lambda_m = 1 \) and the weights are typically optimised using an EM algorithm to maximize the log-likelihood of a development data-set disjoint from that used to train the models themselves [3]. In this paper we significantly increase the number of interpolation weights by estimating a weight for each word context \( (h) \) that occurs in the training data:

\[
P_{\text{interp}}(w | h) = \sum_{m=1}^{M} P_m(w_i | h) \cdot \lambda_m(h). \tag{5}
\]

The caveat is that we will only do this for contexts where we are still able to estimate the weights robustly. We propose using the same training data, that was used to build the component models, to estimate these extra weights. Maximum likelihood estimation would result in a trivial solution so we propose using the leaving-one-out log-likelihood (LL\(^{-LOO}\)) on the training data as the weight optimisation criterion as shown below for bigram model optimisation on a training set of \( N_W \) words:

\[
\sum_{i=1}^{N_W} N(w_i, w_{i-1}) \cdot \log \left( \sum_{m=1}^{M} P_m^{LOO}(w_i | w_{i-1}) \cdot \lambda_m(w_i | w_{i-1}) \right), \tag{6}
\]

where \( P_m^{LOO}(w_i | w_{i-1}) \) is component model \( m \)'s LOO probability estimate. Typically the training data is larger than any held-out development set so we prefer to use it because of the large number of weights we wish to determine. However, we still need to use a held-out development set to determine count thresholds below which weights for contexts

---

* 文脈依存混合重みを用いた単語モデルとクラス言語モデルの混合

エドワード ウィッタッカー（東工大）
will not be optimised. For these contexts we use the global interpolation weights optimised on the development set. For a trigram model we have the following:

\[ P_{\text{interp}}(w_i | w_{i-2}, w_{i-1}) = \sum_{m=1}^{M} P_m(w_i | w_{i-2}, w_{i-1}) \lambda_m(w_{i-2}, w_{i-1}), \]  

where \( \lambda_m(w_{i-2}, w_{i-1}) \) is a two-word context-dependent interpolation weight estimated for each language model. Due to memory access and storage limitations we only estimate weights for two-word combinations that occur a sufficient number of times in the training data. If a two-word context does not occur frequently enough, or at all, we use the one-word context interpolation weight \( \lambda_m(w_{i-1}) \) instead.

4 Experimental work

Here, we present both perplexity and speech recognition results evaluated on test-set1 of the Corpus of Spontaneous Japanese (CSJ) as described in [6]. Bigram and trigram class models were built using class definitions, obtained using an exchange algorithm and by maximizing the log-likelihood of the training data [4], for each of the following three class model types: two-sided symmetric (Equation (1)); two-sided asymmetric (Equation (2)); and one-sided (Equation (3)). For each model type, models with nine different class sizes (\(|C|\)) were built in powers of 2, from 8 to 2048 classes. All models used interpolated Kneser-Ney smoothing [1]. Context-dependent weights were estimated for the model given by Equation (7) only if the context occurred a threshold number of times. The thresholds and global interpolation weights were determined on the development data. An EM algorithm was used to determine the optimal weights which were initialized to be all equal.

Speech recognition experiments were performed using the Julius two-pass speech recognizer with a 30k vocabulary and triphone HMMs that have 3000 shared states with 16 Gaussian mixture components trained on 500 hours of the CSJ corpus. 7M words of acoustic transcriptions were used for language modelling training data with approximately 0.5M words set aside for development. Perplexity and recognition results for the same range of models are presented in Table 1.

Using all models we obtain a 15.7% and 4% relative reduction in PP and WER, respectively, compared to a 10.4% and 3% reduction in PP and WER, respectively, with the ‘traditional’ combination of one word model and one two-sided class model using global weights optimised on development data.

For CSJ the available training data is still relatively small so there is still not enough data available to reliably estimate many interpolation weights. The difference between global weights estimated on the development set and context-dependent weights optimised on the training data is negligible. For example, for the two-sided trigram models, this amounts to only 0.3 absolute perplexity points improvement when context-dependent weights are used. We conclude that most of the improvement comes from the combination of different models rather than the context-dependent weights.

5 Conclusion

A model combination method was presented that used context-dependent interpolation weights to combine a word model and different types of class model. Using class models that capture different dependencies and that use different numbers of classes was found to give improvements in perplexity and word error rate performance over using a single word model or an optimised combination of one word model and one two-sided class model. On a Japanese language Spontaneous Speech task the best model gave a 15.7% reduction in perplexity and a 4% relative improvement in word error rate.

Table 1. PP and WER performance on test-set1 of CSJ. (*) indicates results with global parameters.

<table>
<thead>
<tr>
<th>Language Model</th>
<th>PP</th>
<th>WER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline word trigram</td>
<td>114.0</td>
<td>29.9</td>
</tr>
<tr>
<td>Word and single two-sided</td>
<td>102.2</td>
<td>29.0</td>
</tr>
<tr>
<td>All two-sided symmetric</td>
<td>98.9 (99.2*)</td>
<td>29.0</td>
</tr>
<tr>
<td>All models</td>
<td>96.1 (96.1*)</td>
<td>28.7</td>
</tr>
</tbody>
</table>

References