*Computer Speech and Language* (2001) **15**, 39–53
doi:10.1006/csla.2000.0156
Available online at http://www.idealibrary.com on IDEAL®

# Improved language modelling through better language model evaluation measures

**Philip Clarkson† and Tony Robinson**

*Cambridge University Engineering Department, Trumpington Street, Cambridge, U.K.*

**Abstract**

This paper explores the interaction between a language model's perplexity and its effect on the word error rate of a speech recognition system. Much recent research has indicated that these two measures are not as well correlated as was once thought, and many examples exist of models which have a much lower perplexity than the equivalent $N$-gram model, yet lead to no improvement in recognition accuracy. This paper investigates the reasons for this apparent discrepancy. Perplexity's calculation is based solely on the probabilities of words contained within the test text; it disregards the probabilities of alternative words which will be competing with the correct word within the decoder. It is shown that by considering the probabilities of the alternative words it is possible to derive measures of language model quality which are better correlated with word error rate than perplexity is. Furthermore, optimizing language model parameters with respect to these new measures leads to a significant reduction in the word error rate.



## 1. Introduction

For many years, perplexity (Bahl, Jelinek & Mercer, 1983) has been the measure by which language model quality has been evaluated. There are good reasons for this; it is a simple, well-understood measure that fits into the maximum likelihood framework and it can be computed quickly. However, recent work on language modelling has demonstrated that the correlation between a language model's perplexity and its effect on the word error rate of a speech recognition system is not as strong as was once thought. There are many examples of cases in which a language model has a much lower perplexity than the baseline model, but does not result in a reduction in word error rate, and often results in a degradation in recognition accuracy. This paper investigates reasons for this apparent discrepancy, and describes the development of measures of language model quality that are more strongly correlated with word error rate than perplexity is. The work focuses on the broadcast news task (Pallett & Fiscus, 1997).

The calculation of perplexity is based on the probability that the language model assigns to some test text. If the language model is successful it will assign a high probability to this

†Author to whom correspondence should be addressed: SpeechWorks International, 695 Atlantic Avenue, Boston, MA 02111, U.S.A. E-mail: philip.clarkson@speechworks.com

0885–2308/01/010039 + 15 $35.00/0 

test text, with the result that the language model will have a low perplexity. Thus perplexity is based solely on the probabilities of the words which actually occur in the test text. Previous work (Chen, Beeferman & Rosenfeld, 1998) has investigated ways in which this information can be used to better predict word error rate. In this paper we consider language models with the same perplexity, but which result in different word error rates. We show that merely considering the probabilities of the words which occur in the test text is inadequate to distinguish between them. Thus we show that it is important to also consider the manner in which the remaining probability mass is distributed over the alternative words, which may be competing with the correct word in the decoder of a speech recognition system. We show that including such information leads to measures which are better correlated with word error rate. Finally, we will show how the information from the new measures can be used to select more appropriate interpolation weights for mixture-based language models. Such interpolation weights lead to a small, but statistically significant improvement in word error rate as compared to the original maximum likelihood weights.

## 2. Same perplexity language models

### *2.1. Construction of same perplexity language models*

Previous work (Clarkson & Robinson, 1998) has investigated mixture-based models (where the training text is partitioned according to topic, a language model constructed for each component, and weights assigned to each language model according to the observed style of language) and cache-based models (in which the probabilities of recently occurring words are boosted). Such work has shown that while both models have lower probabilities than the equivalent baseline trigram language model, neither lead to a reduction in word error rate.

If one reduces the amount of training data used to train the mixture- or cache-based language models, their perplexities will be increased. Indeed, if one selects the correct amounts of training data for each language model, it will be possible to generate cache- and mixture-based models that have the same perplexity as the baseline trigram model.

Reducing the amount of training data available to the mixture- or cache-based models is likely to lead to a degradation in recognition accuracy. Therefore, an under-trained mixture- or cache-based language model would be expected to result in a higher word error rate than the baseline model. These models will therefore differ in some way that is important in terms of word error rate, despite having identical perplexities. By investigating the manner in which the models differ it is to be hoped that some light might be shed on the discrepancy between word error rate and perplexity.

Such “same-perplexity language models” were constructed. The baseline language model was a standard back-off trigram model trained on the 130 million word broadcast news corpus (Graff, 1997), with a 65 000 word vocabulary and bigram and trigram cutoffs of 1. The cache-based model was generated by interpolating a static trigram model with a dynamic unigram component trained on the text of the previously-seen portion of the current article. The mixture-based model was built by partitioning the training text into 30 components, and generating a trigram model for each, with a trigram trained on the full set of training data used as an additional component.

The perplexity results are based on the 17 million words of held-out language model text from the broadcast news corpus. Of this, 5 million words are used to estimate appropriate values for the interpolation weights, and the remaining 12 million are used for the

TABLE I. Summary of same-perplexity language models

| Model | % Training data | Perplexity | Word error rate |
|---|---|---|---|
| Baseline | 100 | 134.4 | 37.9 |
| Cache-based | 37 | 134.4 | 39.3 |
| Mixture-based | 42 | 134.4 | 39.3 |

actual perplexity computation. The word error rate results are based on the six shows of the 1996 Hub 4 development test, and were generated by rescoring lattices produced by a simplified version of the 1996 Hub 4 Abbot system (Cook, Kershaw, Christie, Seymore & Waterhouse, 1997). The lattice word error rate (i.e. the word error rate which would result if we chose the path through each lattice with the least errors) for these lattices was 7.0%.

Cache- and mixture-based language models were generated which had the same perplexity as the baseline trigram language model. This was achieved by using only a fraction of the training data for the mixture- and cache-based models, while keeping all other factors the same. As such, a randomly chosen set of articles were removed from the training text. It was found that using 37% of the training data for the cache-based model and 42% for the mixture-based model resulted in models with the same perplexity as the baseline model. Lattice rescoring experiments were conducted using these models in order to determine the models' effect on word error rate. The models are summarized in Table I.

The differences between the recognition accuracy resulting from the use of the baseline model and the mixture- and cache-based models are statistically significant at the 1% level according to the matched pairs sentence segments word error test (Gillick & Cox, 1989).

### *2.2. Estimating the number of words correct*

Consider a function $f_{\mathcal{M}}(x)$ which indicates the probability that a word chosen at random from test text will be assigned a log probability of $x$ by the language model $\mathcal{M}$ (so $\int_{-\infty}^{0} f_{\mathcal{M}}(x)dx = 1$). The value $\mu$ of the mean log probability of the words in the test text can be computed given the values of $f_{\mathcal{M}}(x)$:

$$\mu = \int_{-\infty}^{0} x f_{\mathcal{M}}(x)dx. \tag{1}$$

Since perplexity is based on the mean log probability of the words in the test text $w_1^n$:

$$PP = P(w_1^n)^{-1/n} = e^{-\frac{1}{n}\sum_{i=1}^{n}\log[P(w_i|w_1^{i-1})]} = e^{-\mu}, \tag{2}$$

$f_{\mathcal{M}}(x)$ contains at least as much useful information as the value of perplexity, and possibly somewhat more.

Consider also a function $g(x)$ which indicates the probability that a word with language model log probability $x$ will be recognized correctly. $f_{\mathcal{M}}(x)$ and $g(x)$ can be combined to generate an estimate of the expected number of words correct:

$$E_{\mathcal{M}}(\text{Words correct}) = \int_{-\infty}^{0} f_{\mathcal{M}}(x)g(x)dx. \tag{3}$$

This is potentially a more useful predictor of recognition performance than perplexity. This technique was first investigated by Chen *et al.* (1998), and was used to derive $M$-ref, a measure of language model quality.

Note that this assumes that the function $g(x)$ is constant across all language models. This assumption is necessary, as $g(x)$ can only be estimated based on knowledge of whether words with particular language model probabilities were correctly recognized. Generating this information requires a recognition pass, so if it were necessary for every language model under consideration, one could simply evaluate the word error rate directly, and there would be no need to use techniques to estimate its value.

### 2.3. Estimating the functions $f$ and $g$

The function $f_{\mathcal{M}}(x)$ was estimated by partitioning the probability range into 100 bins which are spaced equally in the log domain. For each language model, the number of words in the test text which have language model probabilities in each bin is computed. The function $f(x)$ was estimated for the baseline trigram model, as well as for the same-perplexity cache- and mixture-based models. Figures 1 and 2 show the resulting functions for the cache- and mixture-based models compared with the baseline trigram model.

The function $g(x)$ was computed individually for each of the language models. The transcription generated using each language model was aligned with the reference transcription using the same dynamic programming algorithm as was used for word error rate scoring, and hence each word in the reference transcription was labelled according to whether it was correctly recognized when each of the language models were used. The probability range was split into 30 equally log-spaced bins[1], and the values of $g(x)$ were then estimated at each of the bin centres according to

$$g(x) = \frac{\text{\#words with LM probs in bin that were correctly recognized}}{\text{\#words with LM probs in bin}}. \tag{4}$$

The resulting functions are displayed in Figures 3 and 4. Figure 3 shows the comparison of $g_{\text{trigram}}(x)$ with $g_{\text{cache}}(x)$, and Figure 4 compares $g_{\text{trigram}}(x)$ and $g_{\text{mixture}}(x)$.

### 2.4. Results

The estimates for the values of the $f(x)$ and $g(x)$ were used to generate estimates for the number of correct words according to the following approximation[2]:

$$E(\text{Proportion of words correct}) = \int_{-\infty}^{0} f(x)g(x)dx \approx \sum_{x \in \text{Bin centres}} f(x)g(x). \tag{5}$$

The estimated proportions of words correct were generated in two ways. Firstly, the estimate was based on the appropriate version of $g(x)$, in order to generate the most accurate value. That is, the expected proportion of words correct using the model $\mathcal{M}$ was calculated using $\sum f_{\mathcal{M}}(x)g_{\mathcal{M}}(x)$. However, since an estimate for $g(x)$ for each language model will not be available in practice, an estimate was also generated using $g_{\text{trigram}}(x)$. So, for the model $\mathcal{M}$, the expected proportion of words correct was $\sum f_{\mathcal{M}}(x)g_{\text{trigram}}(x)$. The results are shown in Table II.

The discrepancy between the actual proportion of words correct, and the values of $\sum f_{\mathcal{M}}(x)g_{\mathcal{M}}(x)$ is due to the fact that $f_{\mathcal{M}}(x)$ is estimated based on the test text, rather than the

[1] Fewer bins were used than in the computation of $f(x)$ since information from a recognition pass was required, as opposed to information from a large, text-only test set, and therefore the data was more sparse.

[2] The "bin centres" referred to are the 100 bin centres used in the estimation of $f_{\mathcal{M}}(x)$. The values for $g(x)$ were taken then from the bin with the closest value.

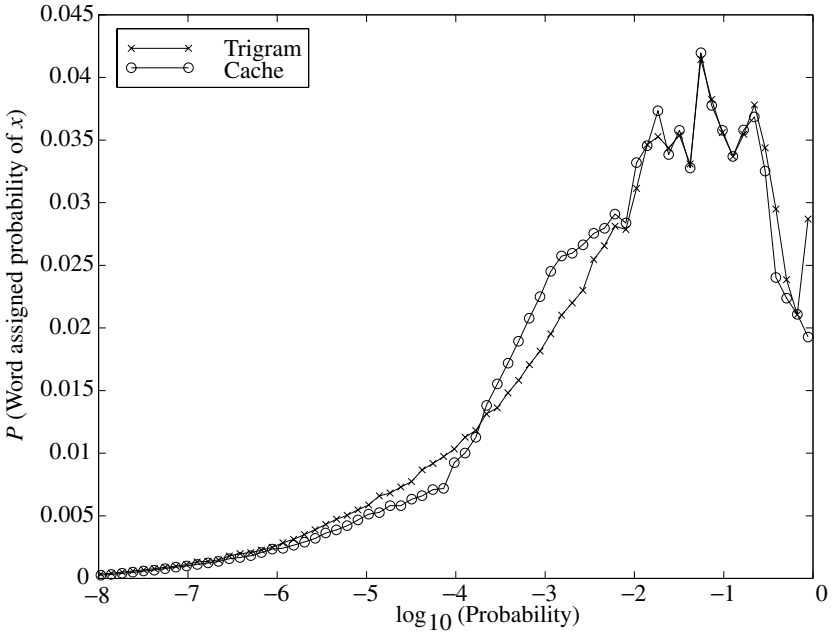


**Figure 1.** Probability distribution graph. Comparison of $f_{\text{trigram}}(x)$ and $f_{\text{cache}}(x)$.

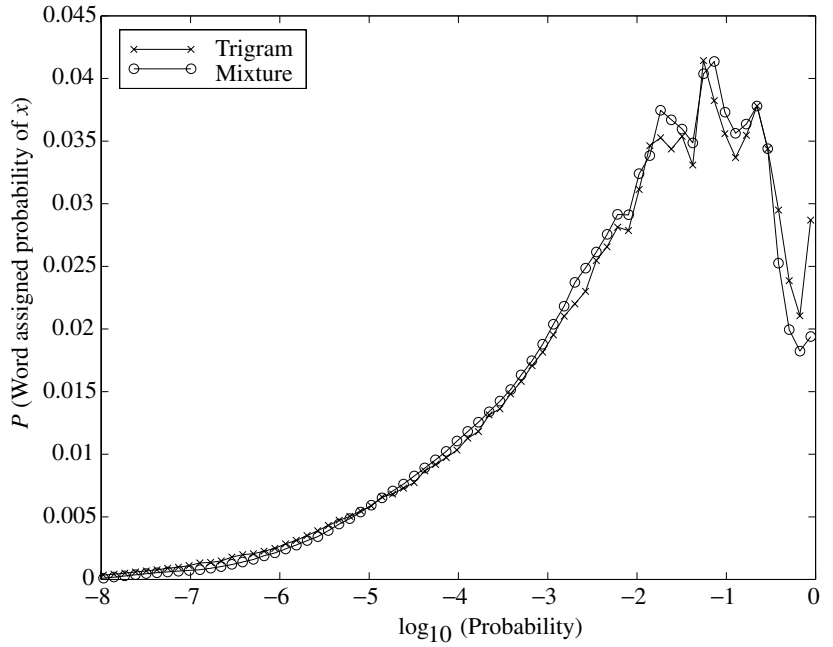


**Figure 2.** Probability distribution graph. Comparison of $f_{\text{trigram}}(x)$ and $f_{\text{mixture}}(x)$.

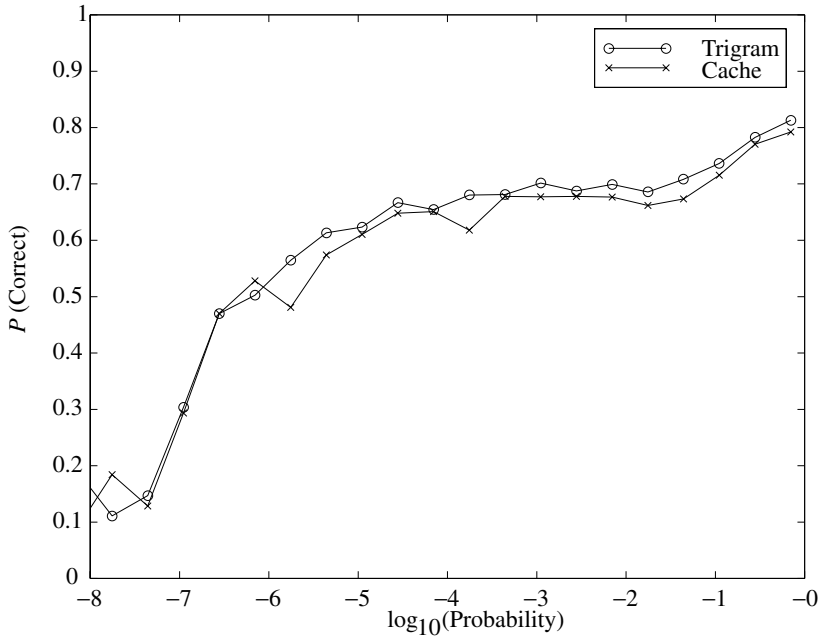


**Figure 3.** Comparison of $g_{\text{trigram}}(x)$ and $g_{\text{cache}}(x)$.

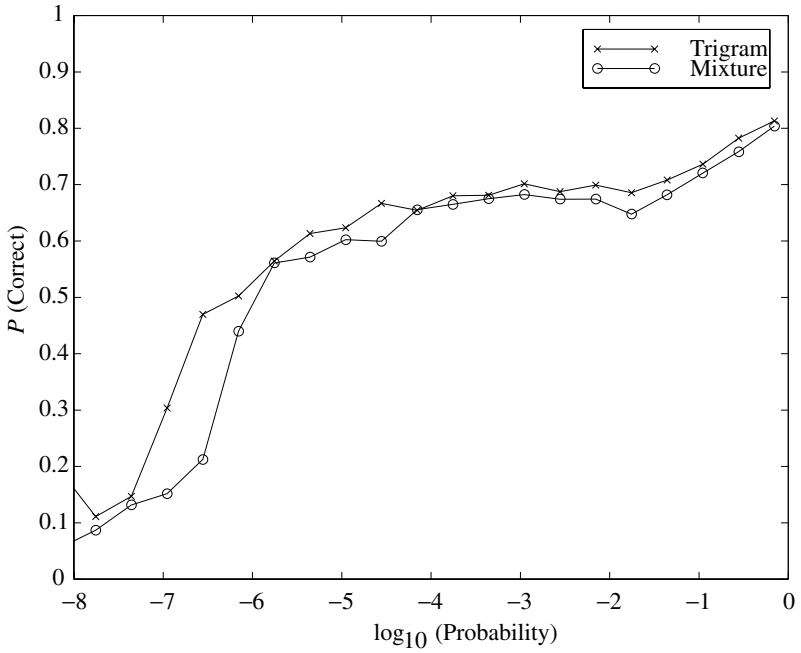


**Figure 4.** Comparison of $g_{\text{trigram}}(x)$ and $g_{\text{mixture}}(x)$.

TABLE II. Estimates of words correct

| Model | Actual words correct | $\sum f_{\mathcal{M}}(x)g_{\mathcal{M}}(x)$ | $\sum f_{\mathcal{M}}(x)g_{\text{trigram}}(x)$ |
|---|---|---|---|
| Baseline | 69.4% | 70.6% | 70.6% |
| Cache | 67.4% | 68.4% | 70.6% |
| Mixture | 67.2% | 68.4% | 70.7% |

reference transcript. The language models have a lower perplexity with respect to the test text than they do with respect to the reference transcription, and this leads to a high estimate for words correct. If the reference transcription were used to generate estimates for the values of $f_{\mathcal{M}}(x)$, then the actual proportion of words correct would be identical to $\sum f_{\mathcal{M}}(x)g_{\mathcal{M}}(x)$.

### *2.5. Discussion*

The first observation to make from Figures 1–4 is that it is not the case that $g(x)$ is constant across all the models. This suggests that it is unlikely to be possible to predict a model's effect on word error rate accurately from its probability distribution curve alone. Indeed, the values of $\sum f_{\mathcal{M}}(x)g_{\text{trigram}}(x)$ in Table II show that the expected values of proportion of words correct generated by $g_{\text{trigram}}(x)$ do not show any difference between the three same-perplexity language models.

This phenomena is shown even more strongly by Figure 2, which shows that the probability distribution curve for the mixture-based model is almost identical to that of the baseline trigram model. These models result in significantly different word error rates, yet there is not sufficient information in the probabilities of the words in the test text to distinguish between them. It is clear then, that the information needed to discriminate between these models is not contained in the probabilities of the words which actually occur in the test text. It therefore seems likely that the information needed to distinguish between the models is contained in the way in which the remaining probability mass is distributed over the *alternative* words, which will compete with the correct word in the decoder of a speech recognizer. It is this observation which motivates the work in the next section.

## 3. Improved measures of language model quality

### *3.1. The use of the whole distribution*

Previously in this paper, language models have been evaluated according to their perplexity. At each point in the test text, the computation of perplexity considers only the probability of the next word in the text. The language model evaluation schemes explored in this section, however, are based on the probability distribution over the whole vocabulary. That is, if the test text is $w_1^n$, then perplexity is based on the values of $P(w_i \mid w_1^{i-1})$, and the new measures will be based on the values of $P(w \mid w_1^{i-1})$ for all $w$. The word $w_i$ which actually follows the word history $w_1^{i-1}$ in the test text will be referred to as the *target word*.

At first glance, it might seem that computing $P(w \mid w_1^{i-1})$ for all words in the vocabulary will require more computation by a factor of $V$ (where $V$ is the number of words in the vocabulary) than simply computing one probability, since $V$ language model probabilities need to be calculated. However, there are some short-cuts which can be applied which mean that this is not the case.

Consider the case of a trigram model, where one is attempting to find the values of $P(w \mid w_1^2)$ for all $w$. The language models used in this work were created using the CMU-Cambridge Statistical Language Modelling Toolkit (Clarkson & Rosenfeld, 1997). In these models the information is stored in a tree structure, with the unigram information at the node and the trigram information at the leaves. Therefore, one can find the position of $w_1^2$ in the bigram layer of the tree, and from there look up the probabilities of all the words $w$ such that the trigram $(w_1, w_2, w)$ exists in the language model. One can then find the position of $w_2$ in the unigram layer, and look up the probabilities of all words $w$ such that the trigram $(w_1, w_2, w)$ does not exist in the language model, but $(w_2, w)$ does. Finally, the probabilities of the $w$ which have not yet been computed can simply be looked up from the unigram layer.

While this process is considerably more computationally expensive than looking up the probability of just the target word, it is much more efficient than simply performing $V$ language model look-ups. In practice, for the software generated for this work, it requires approximately 500 times more computational time to compute the whole distribution for a 65 000 word vocabulary than it does to compute one language model probability.

### *3.2. Language model test set*

In order to investigate the correlation between word error rate and new language model evaluation measures, it is clearly necessary to have a large set of language models upon which to base the experiments.

A set of 50 language models was constructed. These models comprise bigram, trigram, mixture- and cache-based models, which have been trained on either the broadcast news corpus or British national corpus—a very varied corpus consisting of 100 million words of British English (Burnard, 1995). Different quantities of the training corpora were used to train each language model, and various cutoffs were applied. The lattices described in Section 2.1 were rescored using each language model and the resulting word error rate was computed for each. The set of models is summarized in Table III. The table indicates whether the broadcast news (bn) or British national corpus (BNC) was used to train the model, the type of language model (either bigram or trigram), the type of adaptation used (where $C(x)$ represents cache-based adaptation with an interpolation weight of $x$ for the cache component), the proportion of the training data used to train the model, the cutoffs applied and the word error rate. The final three models were based on a two-component mixture model with one component trained on the broadcast news corpus, and the other on the British national corpus. The interpolation weights were fixed, and not intended to reflect the target domain. The adaptation argument $M(x)$ indicates that the mixture weight assigned to the broadcast news component was $x$.

Some interesting points are brought to light by Table III that are worth mentioning in passing. By comparing the word error rates of models 38 and 46, it can be seen that adding a cache component to a bigram language model leads to a reduction in word error rate for models trained on the broadcast news corpus. Similarly, comparing models 6 and 31 reveals that adding a cache component to the broadcast news model with bigram and trigram cutoffs of 20 results in a reduction in word error rate and comparing models 9 and 26 reveals that adding a cache component to a trigram model trained on only a small fraction of the broadcast news data also yields a reduction in word error rate. These results are all in contrast to the results of adding a cache component to the baseline broadcast news model. Furthermore, it can be seen that both cache- and mixture-based adaptation improve recognition accuracy for models trained on the British national corpus (compare models 34 and 36 with model 15).

TABLE III. Summary of language models used to investigate new language model evaluation schemes

| Model | Training corpus | Type | Adaptation | Fraction of training data | Cutoffs | WER |
|---|---|---|---|---|---|---|
| 1 | bn | 3-gram | None | 1.0 | 0 0 | 38.0 |
| 2 | bn | 3-gram | None | 1.0 | 1 1 | 37.9 |
| 3 | bn | 3-gram | None | 1.0 | 2 2 | 38.2 |
| 4 | bn | 3-gram | None | 1.0 | 5 5 | 39.2 |
| 5 | bn | 3-gram | None | 1.0 | 10 10 | 40.0 |
| 6 | bn | 3-gram | None | 1.0 | 20 20 | 40.9 |
| 7 | bn | 3-gram | None | 1.0 | 50 50 | 41.9 |
| 8 | bn | 3-gram | None | 1.0 | 100 100 | 43.1 |
| 9 | bn | 3-gram | None | 0.001 | 1 1 | 52.0 |
| 10 | bn | 3-gram | None | 0.01 | 1 1 | 45.7 |
| 11 | bn | 3-gram | None | 0.1 | 1 1 | 40.9 |
| 12 | bn | 3-gram | None | 0.25 | 1 1 | 39.5 |
| 13 | bn | 3-gram | None | 0.5 | 1 1 | 39.0 |
| 14 | BNC | 3-gram | None | 1.0 | 0 0 | 43.5 |
| 15 | BNC | 3-gram | None | 1.0 | 1 1 | 42.8 |
| 16 | BNC | 3-gram | None | 1.0 | 2 2 | 43.1 |
| 17 | BNC | 3-gram | None | 1.0 | 5 5 | 43.5 |
| 18 | BNC | 3-gram | None | 1.0 | 10 10 | 43.9 |
| 19 | BNC | 3-gram | None | 1.0 | 20 20 | 44.7 |
| 20 | BNC | 3-gram | None | 1.0 | 50 50 | 45.8 |
| 21 | BNC | 3-gram | None | 1.0 | 100 100 | 47.2 |
| 22 | BNC | 3-gram | None | 0.01 | 1 1 | 50.6 |
| 23 | BNC | 3-gram | None | 0.1 | 1 1 | 46.2 |
| 24 | BNC | 3-gram | None | 0.25 | 1 1 | 44.5 |
| 25 | BNC | 3-gram | None | 0.5 | 1 1 | 43.7 |
| 26 | bn | 3-gram | $C(0.1)$ | 0.001 | 1 1 | 50.6 |
| 27 | bn | 3-gram | $C(0.05)$ | 1 | 1 1 | 37.9 |
| 28 | bn | 3-gram | $C(0.1)$ | 1 | 1 1 | 38.0 |
| 29 | bn | 3-gram | $C(0.25)$ | 1 | 1 1 | 38.8 |
| 30 | bn | 3-gram | $C(0.5)$ | 1 | 1 1 | 40.6 |
| 31 | bn | 3-gram | $C(0.1)$ | 1 | 20 20 | 40.5 |
| 32 | bn | 3-gram | Mix | 1 | 1 1 | 38.2 |
| 33 | bn | 3-gram | Mix | 0.42 | 1 1 | 39.3 |
| 34 | BNC | 3-gram | Mix | 1 | 1 1 | 41.8 |
| 35 | BNC | 3-gram | $C(0.1)$ | 1 | 20 20 | 43.8 |
| 36 | BNC | 3-gram | $C(0.1)$ | 1 | 1 1 | 42.0 |
| 37 | BNC | 3-gram | $C(0.1)$ | 0.01 | 1 1 | 49.1 |
| 38 | bn | 2-gram | None | 1 | 1 | 41.7 |
| 39 | bn | 2-gram | None | 1 | 5 | 42.2 |
| 40 | bn | 2-gram | None | 0.01 | 1 | 47.1 |
| 41 | bn | 2-gram | None | 0.5 | 1 | 42.0 |
| 42 | BNC | 2-gram | None | 1 | 1 | 45.2 |
| 43 | BNC | 2-gram | None | 1 | 5 | 45.7 |
| 44 | BNC | 2-gram | None | 0.01 | 1 | 50.7 |
| 45 | BNC | 2-gram | None | 0.5 | 1 | 45.8 |
| 46 | bn | 2-gram | $C(0.1)$ | 1 | 1 | 41.4 |
| 47 | BNC | 2-gram | $C(0.1)$ | 1 | 1 | 44.4 |
| 48 | bn+BNC | 3-gram | $M(0.25)$ | 1 | 1 1 | 38.9 |
| 49 | bn+BNC | 3-gram | $M(0.5)$ | 1 | 1 1 | 38.1 |
| 50 | bn+BNC | 3-gram | $M(0.75)$ | 1 | 1 1 | 38.0 |

These results all indicate that the language model adaptation techniques are of benefit in situations where the baseline model is of low quality or less suited to the target domain.

The extent of the correlation between the new evaluation schemes and word error rate will be evaluated using three correlation coefficients: the Pearson product-moment correlation coefficient $r$, the Spearman rank-order correlation coefficient $r_s$, and the Kendall rank-order correlation coefficient $T$.

The perplexity results reported in previous sections were calculated with respect to the language model test text. However, the language model evaluation schemes which will be described in this section are evaluated with respect to the reference transcription of the broadcast news shows upon which the word error rate scores are based. There are two reasons for this. The most important reason is the potential mismatch between the language model test text and the recognition task. When perplexity is calculated using the language model test text, there is an implicit assumption that this text is representative of the speech which one is attempting to recognize. Due to the nature of the language model test set in this case, this assumption is reasonable. However, one can avoid the need to make it at all by basing the perplexity calculation on the reference transcription. The second reason is a pragmatic one. Since the evaluation of measures based on the whole distribution requires approximately 500 times more computational time than the calculation of perplexity, the opportunity to evaluate them based on a test set of 22 000 words (as opposed to the 17 million word test text) is appealing.

### *3.3. New language model evaluation measures*

#### *3.3.1. Proposed features*

*Log probability (perplexity).* Perplexity has been used as a method of evaluating language models throughout this paper, and in this section it serves as the baseline measure. The correlation between word error rate and perplexity on the test set of 50 language models was evaluated.

*Rank.* Perplexity measures the language model's success according to the probability it assigns to each of the words in the test text. An alternative is to evaluate the language model according to the proportion of words which have a higher probability than the target word at each time point. By so doing, the measure would encode the quality of the target word's prediction relative to the other words with which it will be competing within the speech decoder.

The *rank* of the target word, given a particular history is defined as the word's position in an ordered list of the word probabilities. Thus the most likely word has rank one, the least likely has rank $V$.

For each language model, the rank of each word in the reference file was calculated. Hence the *mean log rank* of each language model was computed, and the strength of the correlation between this measure and word error rate evaluated.

*Entropy.* Given a particular word history $w_1^i$ and a language model, the *entropy* of the probability distribution over the vocabulary[3] is given by

$$H = -\sum_w P(w \mid w_1^i) \log_2 P(w \mid w_1^i). \tag{6}$$

[3]Note that this is different from the test text entropy, which is computed on the whole test set, and is often quoted instead of perplexity.

TABLE IV. Correlation of evaluation measures with word error rate

| | $r$ | $r_s$ | $T$ |
|---|---|---|---|
| Perplexity | 0.955 | 0.955 | 0.840 |
| Mean log rank | 0.967 | 0.957 | 0.846 |
| Mean entropy | −0.799 | −0.792 | −0.602 |
| $L(2^{-5})$ | −0.919 | −0.893 | −0.726 |
| $L(2^{-10})$ | −0.915 | −0.917 | −0.768 |
| $L(2^{-15})$ | −0.833 | −0.817 | −0.640 |
| $L(2^{-20})$ | −0.646 | −0.544 | −0.388 |

Therefore, the entropy is related to the expected value of the log probability given the word history in the following way:

$$E(\log_2 P(w \mid w_1^i)) = -H. \tag{7}$$

Since log probability and entropy are related in this way, and perplexity is based on the *mean* log probability of words in the test text, the measure that was developed was based on the *mean* entropy over the test text.

*Low probability estimates.* The set of 50 language models makes it possible not only to investigate new language model evaluation measures, but also to evaluate previously proposed ones. In particular, in Bahl, Brown, de Souza and Mercer (1989) language models are compared according to the number of words in the test text which receive probability estimates below a certain threshold. The premise is that recognition errors are strongly correlated with very low language model estimates. Therefore, the correlation between word error rate and measures of the form $L(x)$ which measure the proportion of words whose probability estimate is less than or equal to $x$ was investigated.

#### *3.3.2. Results*

The results of each of these new measures are shown in Table IV.

These results show that the mean log rank is at least as well correlated with word error rate as perplexity is, but that the measures based on entropy and the number of low probability estimates (in particular $L(2^{-15})$, the measure used in Bahl *et al.* (1989)) are inferior to perplexity.

The inferiority of the entropy-based measure to perplexity is unsurprising, since the entropy contains only information about the distribution in general, and no information about the target word in particular. However, there is a clear correlation displayed by these results, so some useful information is certainly present in this measure. In Section 3.4 the manner in which this information can be used in a more fruitful way will be investigated.

### *3.4. Combining features*

We now investigate methods of combining the information from some of the measures described above. We begin by examining the correlation between these measures to investigate which might usefully be combined. Then measures which combine the information from the target word's log probability and entropy are constructed and evaluated.

TABLE V. Correlation between language model features

| Feature 1 | Feature 2 | $r_s$ |
|---|---|---|
| Probability | Rank | −0.985 |
| Probability | Entropy | −0.378 |
| Rank | Entropy | 0.381 |

### *3.4.1. Correlation of features*

The following features were selected:

- probability;
- rank;
- entropy;

and the value of $r_s$ for each pair of features was calculated based on their values for each of the words in the test text according to the baseline broadcast news language model (model 2 in Table III).

These results clearly show that there is a very strong correlation between a word's probability and its rank. That is, the two features provide very similar information. Conversely, there seems to be much less correlation between a word's probability and the entropy of the distribution at that point in the test text. Thus, the information provided by these features is, in some sense complementary. Given that both features provide information which is useful in predicting word error rate, it seems that if the information sources can be combined, a superior measure of language model quality would result.

### *3.4.2. Combination of log probability and entropy*

In order to develop measures of language model quality which are better correlated with the word error rate, the information from the probability of the target word and the entropy at each point in the test text was combined.

Since the entropy $H$ is the negative value of the expected log probability of the forthcoming word, the values that were combined were the log probability of the target word $\log_2(P(w_i \mid w_1^{i-1}))$ and the negative entropy $-H(w_1^{i-1}) = \sum_w P(w \mid w_1^{i-1}) \log_2(P(w \mid w_1^{i-1}))$. These values were combined using linear interpolation, both in the log domain, leading to a measure which will be referred to as $C_{\log}(\lambda)$ and after converting back from the log domain, giving a measure called $C_{\text{lin}}(\lambda)$. If the test text is $w_1^n$, then these measures can be expressed as

$$C_{\log}(\lambda) = \frac{1}{n} \sum_{i=1}^{n} [-\lambda H(w_1^{i-1}) + (1-\lambda) \log_2(P(w_i \mid w_1^{i-1}))] \tag{8}$$

and

$$C_{\text{lin}}(\lambda) = \frac{1}{n} \sum_{i=1}^{n} [\lambda 2^{-H(w_1^{i-1})} + (1-\lambda) P(w_i \mid w_1^{i-1})]. \tag{9}$$

The values of these measures were computed for a range of values of $\lambda$. The strength of the correlation between the resulting measures and word error rate was computed, and the results are presented in Table VI.

These results show that combining the information from the two sources leads to language model evaluation measures which are better correlated with the word error rate than either of

TABLE VI. Correlation of combined measures with word error rate

| | $r$ | $r_s$ | $T$ |
|---|---|---|---|
| $C_{\log}(0)$ (Baseline) | 0.966 | 0.955 | 0.840 |
| $C_{\log}(0.05)$ | 0.969 | 0.960 | 0.853 |
| $C_{\log}(0.1)$ | 0.971 | 0.965 | 0.868 |
| $C_{\log}(0.2)$ | 0.971 | 0.964 | 0.863 |
| $C_{\log}(0.3)$ | 0.964 | 0.957 | 0.837 |
| $C_{\text{lin}}(0.001)$ | 0.970 | 0.962 | 0.853 |
| $C_{\text{lin}}(0.002)$ | 0.970 | 0.963 | 0.856 |
| $C_{\text{lin}}(0.01)$ | 0.965 | 0.955 | 0.842 |
| $C_{\text{lin}}(0.02)$ | 0.959 | 0.952 | 0.835 |

the individual measures. In particular, $C_{\log}(0.1)$ performs considerably better than perplexity in this respect. This clearly demonstrates that information concerning the manner in which the probability mass is distributed over non-target words is useful in predicting word error rate.

### *3.5. Application to language model development*

In the mixture-based language model, the interpolation weights assigned to each component are selected to maximize the likelihood (and hence to minimize the perplexity) of previously seen text. This has typically led to models which have considerably lower perplexities than the baseline trigram model, but no decrease in word error rate.

This section has described the development of measures of language model quality which correlate better with word error rate than perplexity does. Since the ultimate aim of mixture-based models is to reduce word error rate, the interpolation weights should be chosen with this in mind. Therefore, we attempt to choose interpolation weights which are optimized with respect to our new measures.

The probability estimate from the mixture-based model is simply a linear combination of the probability estimates from a set of component models:

$$P(w_i \mid w_{i-2}^{i-1}) = \sum_{j=0}^{k} \lambda_j P_{\text{model } j}(w_i \mid w_{i-2}^{i-1}) \tag{10}$$

where "model 0" is the full language model, and $k$ represents the number of components into which the training text is clustered.

The aim, therefore, is to select interpolation weights $\lambda_j$ in order to maximize a more appropriate measure. In this case, we aim to maximize $C_{\log}(0.1)$. This technique was applied to generate new interpolation weights for mixture-based models based on 30 mixture components trained on both the broadcast news corpus and the British national corpus. In both cases, supervised and unsupervised adaptation were applied. Lattice rescoring experiments were carried out using the resulting interpolation weights. The results are presented in Table VII, and are compared with the results of using the conventional maximum likelihood weights.

These results show that the new weights chosen to maximize $C_{\log}(0.1)$ perform consistently better than the the old maximum likelihood weights. While the difference in performance is small, the overall difference between the word error rate optimized and maximum

TABLE VII. Comparison of word error rates achieved by maximum likelihood and word error rate optimized interpolation weights

| | | Weights | |
|---|---|---|---|
| Training text | Adaptation | Maximum likelihood | WER optimized |
| Broadcast news | Unsupervised | 38.2% | 38.1% |
| Broadcast news | Supervised | 38.0% | 37.9% |
| BNC | Unsupervised | 42.3% | 41.9% |
| BNC | Supervised | 41.8% | 41.6% |

likelihood weights is statistically significant at the 1% level according to the matched pairs sentence segments word error test (Gillick & Cox, 1989).

## 4. Conclusions and future work

This paper has investigated the shortcomings of perplexity as a predictor of a language model's effect on word error rate, and has proposed alternative measures. Language models with identical perplexities but which result in significantly different word error rates have been developed. By investigating the manner in which these models differ, it has been shown that merely considering the probabilities of the words which occur in the test text is insufficient to distinguish between them. Instead, relevant information is contained in the probabilities of the alternative words which will compete with the correct word within the decoder. By also considering this information, it was shown that it is possible to derive measures of language model quality which are considerably better correlated with word error rate than perplexity is. It was further shown that the recognition performance achieved using mixture-based language models was increased when the mixture weights were chosen to optimize these new measures rather than perplexity.

There are many ways in which this information about the probabilities of alternative words could be used in measures of language model quality, and only a few of these have been investigated in this paper. There is therefore a good deal of scope for developing measures which correlate even more strongly with word error rate. Such improved measures will be useful in themselves, and are likely to lead to further-improved language models.

The authors wish to thank Roni Rosenfeld, Kristie Seymore and Stan Chen of Carnegie Mellon University for many useful and informative discussions.

## References

Bahl, L. R., Brown, P. F., de Souza, P. V. & Mercer, R. L. (1989). A Tree-based statistical language model for natural language speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **37(7)**, 1001–1008.

Bahl, L. R., Jelinek, F. & Mercer, R. L. (1983). A maximum likelihood approach to continuous speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **5(2)**, 179–190.

Burnard, L. (1995). *Users Reference Guide for the British National Corpus*, Oxford University Computing Services.

Chen, S., Beeferman, D. & Rosenfeld, R. (1998). Evaluation metrics for language models. *Proceedings of the ARPA Workshop on Human Language Technology*, Lansdowne, Virginia.

Clarkson, P. R. & Rosenfeld, R. (1997). Statistical language modeling using the CMU-Cambridge toolkit. *Proceedings of the European Conference on Speech Communication and Technology*, volume 5, Rhodes, Greece, pp. 2707–2710.

Clarkson, P. R. & Robinson, A. J. (1998). The applicability of adaptive language modelling for the broadcast news task. *Proceedings of the International Conference on Spoken Language Processing*, Sydney, Australia.

Cook, G. D., Kershaw, D. J., Christie, J. D. M., Seymore, C. W. & Waterhouse, S. R. (1997). Transcription of broadcast television and radio news:The 1996 Abbot system. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 2, Munich, Germany, pp. 723–726.

Gillick, L. & Cox, S. J. (1989). Some statistical issues in the comparison of speech recognition algorithms. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 1, Glasgow, UK, pp. 532–535.

Graff, D. (1997). The 1996 broadcast news speech and language-model corpus. *Proceedings of the ARPA Workshop on Human Language Technology*, Chantilly, Virginia.

Pallett, D. & Fiscus, J. (February 1997). 1996 Preliminary broadcast news benchmark tests. *Proceedings of the DARPA Speech Recognition Workshop*, Chantilly, Virginia.

*(Received 16 June 2000 and accepted for publication 13 September 2000)*