

Using Discriminative principles for recognising City Names

Gies Bouwman, Louis Boves

A²RT, Department of Language and Speech

University of Nijmegen, The Netherlands

{G.Bouwman,L.Boves}@let.kun.nl

<http://lands.let.kun.nl/>

Abstract

In this paper we address the problem of mismatch in train and test conditions. Counter intuitive as it may seem, we do this by employing a particular element from the well-known training paradigm of Minimum Classification Error training. Rather than recognising a sentence according to maximum likelihood, we examine a number of likelihood ratio-based word score techniques in order to rescore and resort N-best lists.

Experiments for a Dutch city name recognition task did not lead to improved recognition performance. Analysing the results however, we see a number of promising handles for more succesful attempts in the future. We find cues that the information modelled in antimodels is not only useful for keyword spotting and confidence measure assessment, but may be valuable for decoding as well.

1. Introduction

Error analysis, both automatic and manual, is a powerful technique to guide the process aimed at the improvement of automatic speech recognisers (ASRs). All errors have one feature in common: the recognition models do not represent the erroneously recognised tokens well enough. Two explanations are available for insufficient performance of the models:

1. Not enough training data was available and/or the available material is not employed in an optimal way.
2. There is a structural mismatch between training and operational conditions.

In this paper we will not address the case of insufficient amount of training material, despite the fact that clever addition of material may make a substantial contribution to the solution of remaining problems. In order to tackle the first problem, discriminative training techniques such as Minimum Classification Error (MCE, see [1]) training were introduced as a powerful means to learn to separate the acoustic classes optimally, rather than to estimate the ‘true’ distributions that lie underneath these classes. In MCE this is accomplished by taking an initial set of ML-trained HMMs and defining a loss function over the training data; each incorrectly classified train sample contributes to increase of the loss. The acoustic parameters of all HMMs are adapted such, that total loss is minimised. Classification, loss computation and model adaptation are iterated until loss-decrease drops below some convergence threshold. Model parameter estimation is now directly related to the ultimate goal of the models, viz. to make the optimal classification decision. Substantial accuracy improvements were reported in [1]. Moreover, discriminative

training may yield an especially effective way to tackle the problem of recognising highly confusable words.

However, the problem of mismatch in training and operational circumstances is not addressed by MCE. In order to learn more about this problem, we focus on a particular and important element of the MCE approach, viz. the one that concerns the misclassification measure. This measure is the difference between the likelihood of the target states and the best incorrect competing states. A sample token of a target (sub)word unit is ‘lost’ if this measure is negative, i.e. the sample is more likely to belong to some competing unit than to the target (see [1]). This likelihood evaluation and comparison is actually highly similar to a log-likelihood ratio (LLR) test, as often applied in the context of confidence measures, keyword spotting and rejection (see [2],[3],[4],[5]). Drawing the parallel one step farther, one could also speak of a ‘loss’ situation when a word is misclassified, for example, due to train-test mismatch.

In addition to substantial similarities, there are also differences between the MCE misclassification metric and word level LLRs. In MCE training, the label or ‘target model’ of the misclassified sample is known, so we know for which states to compute the likelihood score difference. To compensate for this lack of knowledge, in case of classification one ought to use some other hypothesis. One such alternative hypothesis could be that the sample belongs to ‘any other but’ the most likely class. A second difference is the level where scores are computed. For the misclassification measure in MCE training, it is not necessarily the word level at which the error is minimised, while for word level LLRs it is. Therefore, in this paper we will examine some of the many different methods to propagate state level LLRs to word level scores.

Models trained under an MCE criterion have a strong discriminative nature and relate directly to confusability of modelled units. ML-trained models maximise the likelihood of the train material and relate strongly to the specific conditions of training, which may well be different from test conditions. Since minimum confusion is the goal we aim for, it appears attractive to use a misclassification measure for classification/decoding. In the present study we investigate the circumstances under which an LLR-derived measure is a better classification/decoding criterion than a classical likelihood score. This idea could contribute to the solution of three problems that pertain in large vocabulary, high perplexity ASR tasks:

1. Mismatch in training and testing conditions, e.g. caused by poorly modelled background and channel noise and non-speech events, is less damaging, since the anti-scores and the recognition scores suffer in a similar way from mismatch,

2. Words which are easily confused with a candidate hypothesis are modelled in the anti-hypothesis. This contributes to the selection of a final candidate on the basis of the most distinctive properties rather than overall likelihood.
3. The scores can be used directly as confidence scores.

With this research we hope to learn how we can make use of the information modelled by antimodels for optimal classification. Two questions are addressed specifically, viz. the contribution of a normalisation of the phone-based LLR scores, and the methods with which frame scores are combined to phone and word scores.

The paper is set up as follows. The next section elaborates on the way we used LLR-derived scores for classification. Section 3 gives further details about our experiments. In Section 4 we present our results, which will be discussed in Section 5. In the sixth section, we summarize our approach and enumerate the most valuable lessons we learned.

2. LR rescaling

2.1. Experimental setup

To test the ideas described above, we set up a number of experiments for a city name recognition task. The most radical approach would involve implementing an ASR system with models optimised to an LLR criterion and to compute LLR scores during search and decoding as well, as proposed in [6]. A result is that the recognisers may become very different systems for each of the word level combinations of LLR scores we would like to investigate, making an interpretation of the results quite complex. To keep our analysis straightforward, we chose to design a procedure that is above all tractable. Our approach was to take the N-best results of a baseline ML-trained system and to rescore and reorder the list with several kinds of LLR-derived word scores. The LLR scores were computed with the procedure described in the next section.

2.2. Rescaling procedure

The ASR system was configured to generate word graphs that contain the most likely recognition results. Next, a post processor took these word graphs and generated N-best lists. Then we pruned these lists in two steps. (1) Of all hypotheses that pertain to the same city name, we preserve only the most likely one. (2) Of the resulting list, we discard all entries from position 6 and higher. Our baseline evaluation reports how often the correct solution is at the first position of these lists (ER_{1best}) and the frequency that it occurs in this list at all (ER_{5best}). It needs no explanation that the latter is a ceiling value for the ER_{1best} of a procedure that reorders the lists.

Next, we computed an LLR-derived score for each member of the 5-best lists in the following way. In a Viterbi alignment, we obtained the five exact state-level segmentations of the incoming speech signal. For each feature vector \mathbf{x}_i assigned to state S_j we then computed the difference of the log probability scores of the corresponding target and anti model, $LLR(x_i|S_j)$.

$$LLR(x_i | S_j) = \log P(x_i | S_j) - \log P(x_i | \hat{S}_j) \quad (1)$$

where \hat{S}_j is the anti-hypothesis of S_j . In section 3.5 we will detail the way we defined and trained our antimodels. One of the things we investigated was whether it helps to normalise for average μ and variance σ of the LLR scores per phoneme type.

$$LLR^*(x_i | S_j) = \frac{LLR(x_i | S_j) - \mu(LLR(S_j))}{\sigma(LLR(S_j))} \quad (2)$$

We tested the following 4 combination rules of combining frame scores to word scores. For convenience, we use the symbol LLR to refer to both the non-normalised frame scores of (1) and the normalised scores of (2).

$$\mathbf{Mea} \quad CM(W) = \frac{1}{0.4} \ln \left[\frac{1}{Np(W)} \cdot \sum_{k=1}^{Np(W)} e^{\left(\frac{0.4}{Nf(P_k)} \sum_{j} LLR \right)} \right] \quad (3)$$

is the emphasised phone-level averaged LLR scores, where $Np(W)$ is the number of phones in W and $Nf(P_k)$ is the number of frames assigned to the k^{th} phone of W , P_k . We chose this combination formula, because it has been quite effective for verification purposes (see [5] for more details).

$$\mathbf{Msa} \quad CM(W) = \sum_{k=1}^{Np(W)} \left[\frac{1}{Nf(P_k)} \sum_j LLR \right] \quad (4)$$

is the summation of the phone-level averaged LLR scores.

$$\mathbf{Mss} \quad CM(W) = \sum_{k=1}^{Np(W)} \left[\sum_j LLR \right] \quad (5)$$

is the sum of the phone-level summed LLR scores.

$$\mathbf{Maa} \quad CM(W) = \frac{1}{Np(W)} \cdot \sum_{k=1}^{Np(W)} \left[\frac{1}{Nf(P_k)} \sum_j LLR \right] \quad (6)$$

is the average of the phone-level averaged LLR scores.

Finally, we add up all word scores of a sentence, augmented with the corresponding class bigram likelihood scores. These scores are used to reorder the 5 best list.

3. Remaining method features

3.1. Test corpus

The test material used for our experiments is a subset of the Dutch Directory Assistance Corpus (DDAC2000) [7]. The recordings are from a real nation-wide directory inquiry service, in which callers were prompted to specify name of the city in which they requested a listing. The total corpus used for the research described in this paper contains 10,954 utterances. In 96.9% of the utterances, the caller mentions a city name or says that (s)he doesn't know it. 7.3% of the utterances contain at least one OOV word.

Recordings were made from the public switched telephone network. The signal was sampled at 8 kHz and stored in a-law format. Acoustic pre-processing comprised extracting 14 MFCCs (c0..c13) and their first-order derivatives from 16 ms Hamming windowed frames, with a 10 ms shift.

3.2. Acoustic models

Acoustic models were trained on 42,101 short utterances of the Dutch Polyphone database [7]. The HMM set consists of 37 tristate monophone models, one tristate noise and two single state models: one for silence and one for garbage speech. In each state acoustic variance is modelled by a mixture pdf of maximally 32 Gaussians.

3.3. Lexicon

The lexicon contains all 2377 Dutch city names, 12 province names, 3 garbage tokens of different length, 1 non-speech noise symbol, 2 entries for filled pauses, 3 multiword expressions for ‘I don’t know’ and 4 frequently used context words.

3.4. Language model

We did our experiments with a Continuous Speech Recognizer that uses probabilistic language models, even though a grammar might have been more suitable for this task. Nevertheless, to illustrate that this task is not ‘plain’ isolated word recognition, Table 1 shows the number of words per utterance. Note that noise is considered as a word as well.

#words/ utterance	percentage of corpus	cumulative
1	62.5%	62.5%
2	28.0%	90.5%
3	5.4%	95.9%
≥4	4.1%	100.0%

Table 1: number of words per utterance.

To steer the process of selecting lexicon items during the computation of the word graphs, we trained a category bigram language model with categories for city names and province names. The within-category unigram for city names was estimated on the number of streets of each city in the Dutch zipcode book. The province name is mentioned by the caller in the exceptional case that a city’s name is not unique and needs disambiguation. The unigram distribution of each member of this category was estimated on the total number of streets of all cities with ambiguous names in that province.

3.5. LLR models

This section describes how we defined and trained target and anti-models to compute the LLR based frame scores.

Since the anti-models are being used for minimising the number of confusions, they should reflect the phonemes with which they are easily confused. To determine the set of most confusable phones, we used the recogniser to segment the training corpus. Each phone was then scored against the ML-trained models for all other phones. In this way, we obtained a phoneme confusion list for each phone type. From that list all ‘unavoidable’ confusion pairs (i.e., confusions between phones that have virtually identical spectra) were removed. We used the resulting lists in the following procedure.

The alignment described in the previous paragraph was used to compute a state-level likelihood score for each feature vector. Next, we computed phone scores by averaging frame scores at phone level. The tristate target models were subsequently trained on the best scoring 95% of all tokens, in order to reduce risk that mislabelled train tokens are included. During training, the state level segmentation of the material was

kept unaltered. Then we trained single state anti-models on the best scoring 20% tokens of the 8 most confusable phoneme types. This procedure cannot be followed for garbage speech and silence. We decided to make them the antimodel of each other. Target and anti-models are HMMs, each having mixture pdfs of maximally 32 Gaussians per state.

3.6. Start/end pointing

From the test material, we know that many of the recordings have extremely long silence tails trailing the speech. For methods **Mea**, **Msa** and **Maa**, which compute the mean score of a segment, this has severe influence on the comparability of the scores.

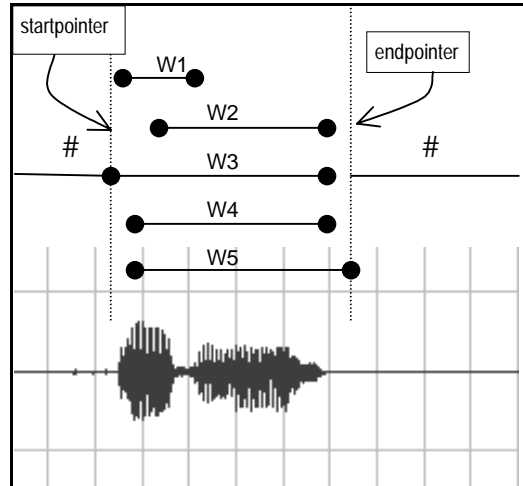


Figure 1: End pointing

To illustrate the problem, consider the following example. If a short word hypothesis has a very good local match, like word W1 in Figure 1, it may get a high average speech score. But the same hypothesis assumes silence right after the word, and this is not the case. However, the LLR for silence may not punish very hard, since there is indeed a lot of silence from the end of W1 until the last frame of the utterance. Therefore, we determined a start and end pointer for each utterance. The pointers correspond to the first and last frames of the time interval spanned by the 5 hypotheses, as illustrated in Figure 1. We compute the silence scores within these boundaries separately from those outside. This guarantees that all hypotheses get identical scores on the segments where they agree that it is silence.

4. Results

This section presents the preliminary results that we obtained. Table 2 shows the results of our baseline system that uses Maximum Likelihood as the classification criterion.

System	ER _{1best}	ER _{5best}
Base	45.8%	32.1%

Table 2: Error rates of the system that generated the N-best lists.

Table 3 shows two error rates for each of the 2 x 4 LLR-derived word scores. The column ‘total’ displays the ER_{1best} for the concerned system. The column ‘common’ shows the

percentage of utterances that were incorrectly classified by both the baseline and the corresponding rescoring system.

comb.	LLR*		LLR	
	common	alone	common	alone
Mea	42.8	59.3	41.9	57.5
Msa	39.4	55.2	39.4	49.8
Mss	38.7	54.1	38.4	49.3
Maa	41.5	62.1	42.1	59.5

Table 3: ER_{1best} of all rescoring systems and common ER_{1best} with the baseline system.

5. Discussion

Starting with the error rates of the four rescoring systems, we see that all have a counter-productive effect on the N-best lists when compared to **Base**. LLR-rescoring has not improved recognition performance.

Looking at the four combination strategies, **Mss** outperforms the other three. Like the **Base** system, this system adds up all scores without any temporal normalisation. In [3] and many other studies relating to confidence measures, it was shown that temporal normalisation is useful. From our results, however, we deduce that time normalisation may be helpful in the case we want to compare to an absolute threshold. When comparing across different hypotheses, however, normalisation in the time domain has nothing but harmful effects.

Just as interesting is the comparison between the scores for (normalised) LLR* and (raw) LLR for the sub-columns ‘alone’ in Table 3. With all four combination techniques the raw LLR show a lower error rate. Apparently the classification measure is not improved by a normalisation of the frame scores for the average mean and variance. We deduce that frame/phone scores that vary over a larger range do indeed reflect a stronger confidence than scores for phones that have a lower mean and variance. This confirms the findings in [4], that some phone confidence scores have more discriminative power than others.

By comparing the common error percentage of two classifiers with each of the individual systems, one can see if a performance increase can be expected from a combination strategy. The set of utterances in the that have no correct solution in the 5-best list is a subset of the set of common errors of **Mss** and **Base**. And since this intersection is a subset of the 1-best errors of **Base**, we find a justification for the following statement. In more than half of the cases where the correct solution is present in the 5-best list but not selected by the baseline system, **Mss** selects the correct solution. This suggests that there is quite some valuable information in the sum of all anti-scores.

In order to study the possibilities of a combination, we have carried out a preliminary error analysis of **Base** and **Mss**. We found that in cases that plain word likelihood scores are more or less the same, anti-models may give a decisive clue about whether one or more phone hypotheses are very unlikely for the concerned segment of speech. This finding should allow us to develop effective heuristic approaches to merge the information from **Base** and **Mss** decoding.

6. Conclusion

In this paper we proposed and evaluated a rescoring framework for N-bestlists, based on discriminative treatment of candidate hypotheses.

Although the first results did not lead to an increase in recognition performance, we have learned two valuable lessons that may be helpful to lead us to successful results of future attempts.

Lesson 1:

Measures that have been successful for verification purposes are not necessarily optimal for classification. ‘Smart tricks’ like temporal and score normalisation (as in [3] and [5]), are counter productive when making cross-hypothesis comparisons, instead of comparing to an invariant threshold.

Lesson 2:

In case of train-test mismatch there is a situation that the (sub)words have a structural deviation from the prototypes in the train conditions. Since antimodels are designed to capture most of what is not prototypical, this may be an important reason that our attempts thus far have not been successful for mismatch situations as we were thinking they would be.

7. Acknowledgements

This research is supported by the EC under the IST-HLT Programme.

8. References

- [1] E. McDermott, *Discriminative Training for Speech Recognition*, Ph.D. Thesis, Waseda Japan, 1997, http://www.hip.atr.co.jp/~mcd/mcd_thesis.ps.gz
- [2] C.-H. Lee, *A Unified Statistical Hypothesis Testing Approach to Speaker Verification and Verbal Information Verification*, Proc. COST250, Rhodes, Greece 1997, pp. 63-72
- [3] G. Bernardis & H. Bourlard, *Improving Posterior based Confidence Measures in HMM/ANN Speech Recognition Systems*, Proc. ICSLP '98, Sydney, vol. 3, pp. 775-778
- [4] G. Bouwman et al., *Weighting Phone Confidence Measures for ASR*, Proc. COST249, Ghent 2000, pp.59-62
- [5] G. Bouwman et al., *Effects of OOV Rates on Keyword Rejection Schemes*, to appear in Proc. Eurospeech 2001
- [6] E. Lleida & R.C. Rose, *Utterance Verification in Continuous Speech Recognition: Decoding and Training Procedures*, IEEE Transactions on Speech and Audio Proc., Vol. 8, No. 2, March 2000, pp. 126-139
- [7] J. Sturm et al., *Impact of speaking style and speaking task on acoustic models*, Proc. ICSLP 2000, Beijing, pp. 361-364