Utterance Verification based on the Likelihood Distance to Alternative Paths

Gies Bouwman and Lou Boves

Department of Language and Speech, P.O. Box 9103,
6500 HD Nijmegen, The Netherlands
{G.Bouwman, L.Boves}@let.kun.nl
http://lands.let.kun.nl

Abstract. Utterance verification is the process where one tries to automatically reject incorrectly recognised utterances, while accepting as many correct results as possible. For this purpose the probability of an error is often estimated by a one-dimensional confidence measure [3], [4], [5], [2]. In this paper we take a closer look at incorrect classification. We argue that errors stem from a number of possible causes and that this observation must be reflected in the design of the utterance verifier.

Therefore, we try to detect both out-of-vocabulary (OOV) word errors and in-vocabulary substitution errors. To this end, we compute confidence measures based on the distance between the likelihood of the first best output and two alternative hypotheses: one corresponding to the second best output, the other to the most likely free phone string.

The paper reports on experiments on spoken Dutch city names for a nationwide directory assistance application. We checked the validity of our ideas by comparing the lowest possible Confidence Error Rates [9]. The results show that at least a 10% CER reduction can be achieved by using a classification and regression tree instead of a linear combination of the cues with a threshold value.

1 Introduction

The need for using confidence measures in ASR applications no longer requires an explicit motivation. Their usefulness is obvious and the issue of rejecting incorrect hypotheses has become a research area on its own. At the moment, we are exploring confidence measures within the scope of the SMADA project [1]. In this project, we study the practical implications of developing a system for nationwide directory assistance (DA). One of the subtasks is to recognise one out of 2,369 possible Dutch city names or the expression for 'I don't know'. Although we have an estimate of the prior probabilities of the city names, it still is a high perplexity task in which most of the information for the eventual ASR output is to be extracted from the acoustic signal. Given the constraints of speaker independent modelling and limited acoustic bandwidth, we face recognition error rates that exceed 10%, and that will not be easy to reduce substantially in a maximum posterior probability decoder. In order to enable a user-friendly dialogue in an automatic DA application with a 10% error rate in the ASR, it is mandatory to automatically reject the least reliable recognition results. Another goal for which we need to identify utterances that were most probably misrecognised is automatic update of acoustic and language models on the based on speech that is recorded during the actual operation of the service. However, before we can discuss our approach to utterance verification in more detail, we must first take a closer look at the nature of errors and their causes.

Many studies (among others [2]) have presented analyses to identify the origin of speech recognition errors. It is necessary to distinguish at least two different types of errors:

- the input speech is (partly) not modelled, for instance because it contains OOV words or word sequences that were not forseen in the grammar. As long as special measures, such as a garbage model, are not taken, "in domain" interpretation of those utterances is doomed to lead to one or more insertion errors.

- the input speech is modelled, but an alternative (incorrect) hypothesis happens to obtain a higher likelihood score. In this case a substitution error occurs.

Of course, deletion errors can also occur, if the likelihood of the garbage model exceeds the score of all in-vocabulary words. However, for the task under analysis in SMADA, deletion errors are less important. In the dialogue they will be handled by a reprompt. In automatic model update it is probably wise not to include utterances in which no vocabulary word was recognised.

Because the causes of recognition errors can be many and varied, it would be somewhat surprising if a single indicator of the reliability of an output is sufficient to flag (virtually) all errors. Therefore, some investigators have attempted to combine multiple confidence measures [2], [4], [5], [3], [6]. Such combinations can take many different forms. In this paper we compare the power of a linear combination and a CART-like procedure.

In the next section we propose two measures to detect the two kinds of errors. This section also describes our general system architecture and the material used to train and test the classifiers. In Section 3 we present more details about the different definitions of the distance measures that we used as atomic indicators of recognition confidence. We also explain the CART procedure that we compared with a linear combination of atomic confidence measures. Section 4 presents the results in terms of the evaluation metric "Confidence Error Rate" and the Receiver Operation Characteristic (ROC) curves. Next, in the discussion section we will give our interpretation of the results. Finally, Section 6 describes the main conclusions and perspectives for future work.

2 Method

2.1 Path distance

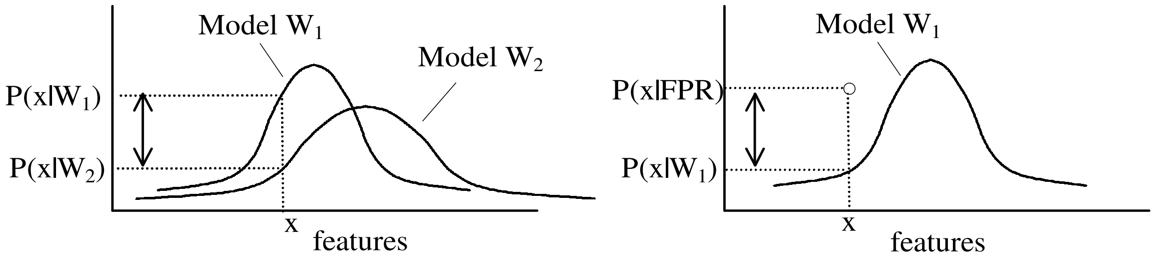

Fig. 1. Left part: likelihood distance between best candidate and free phone string. Right part: distance between best candidate and runner-up.

Figure 1 illustrates the idea of our approach in a simplified manner. Suppose that the gaussian shapes symbolise models of valid words in a one-dimensional feature space. The left part of the figure shows a situation where P(x | W_1) / P(x | W_2) is much greater than 1. This can be interpreted that a substitution error with W_2 is not likely to occur and the classification of x as W_1 is probably correct. In other words, the likelihood ratio of the two best hypotheses can serve as a confidence measure to detect substitution errors.

However, incorrect classification may also be due to OOV speech. In that case this confidence measure is not appropriate. The right part of Figure 1 shows a solution for this problem. The small circle corresponds to the likelihood of the optimal phone string for the feature input, obtained with

free phone recognition (FPR). Its score can serve as a normalisation coefficient for the likelihood corresponding to W_1. In this way we can not only deduce that the word likelihood P(x | W_1) is higher than any other word, but also that the free phoneme models are able to maximise the likelihood much more. Related to FPR, P(x | W_1) is small. The observation x may result from the presence of one or more OOV word(s) and the classification of x as W_1 is unreliable.
In the following subsections we will elaborate on our general system architecture, our training and test material and some other invariant elements of our design.

2.2 Architecture of the utterance verifier

We implemented our utterance verification method as a two-pass procedure.

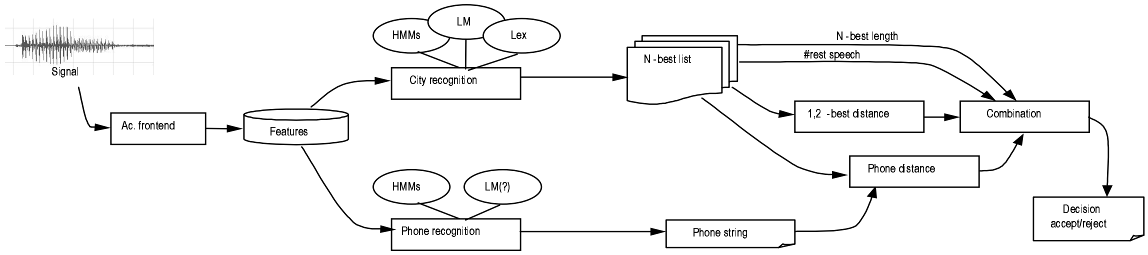

Fig. 2. System architecture

Figure 2 shows that acoustic features are first extracted from the incoming signal. In parallel to the recogniser generating an N-best city name list, the speech input is also decoded in terms of the optimal phone path. In the following steps, we compute four cues to decide whether the best solution is to be rejected or accepted:

1. the length of the original N-best list (> 0, $\leq$ N);
2. the number of frames assigned to other speech models than those of the recognised city name, like the garbage model or alternative city names;
3. the distance between the paths of the first and second best city name hypotheses;
4. the distance between the path of the best hypothesis and the path of phoneme recognition.

In our opinion, these cues have mutually independent predictive power. In the final step, we use either a linear combination (LC) or a classification and regression tree (CART) [7] to come to a decision. The linear combination combines the four measures into a one-dimensional confidence measure that allows one to reject (or accept) utterances based on some threshold value. The CART approach keeps the four dimensions separate. We will return to this issue in the experiments described in Section 3.

2.3 Material

Our material consists of the city name utterances of the DDAC2000 corpus [8], which was collected in a Wizard-of-Oz setup. Callers were prompted to say for which city they wanted a directory listing. The speech recordings were stored in 8 kHz 8-bit A-law format. Acoustic preprocessing comprised 16 ms windowing with 10 ms frame shift and extracting 14 MFCCs (c0-c13) and deltas as the feature vectors. This paper reports on 3 corpora: a training corpus with 4.5 hours of speech (25k utterances), a 142 minutes development corpus (11k utterances) and a test corpus with 144 minutes of speech (11k utterances).

2.4 Models

For both city name and phoneme recognition we used the same set of acoustic models. These were 37 monophone HMMs with additional models for noise, silence and general speech. In each state, the acoustic features are modelled by a mixture pdf of maximally 32 Gaussians. The lexicon for the ASR contains 2,369 Dutch city names, 12 province names, the multi-word expression 'I don't know', 3 frequently used context words and 3 garbage 'words'. Garbage words repeat the general speech model 3, 6 or 11 times.

We used a category-based bigram to estimate language model probabilities. Among others, there were categories for the city names, the province names and garbage speech. Within the category for city names, we estimated the frequency distribution from the number of streets per city, according to the Dutch zip code book. The category bigram part of the model was trained on our training corpus. The perplexity as assessed on the test set amounts to 204.5.

2.5 Content words

In our recognition experiments, the city names (or the expression 'I don't know') are the content words. These words convey the relevant information for the dialogue manager in this part of the interaction. Therefore, we evaluate recognition and verification only at that (semantic) level.

As already mentioned, our speech recogniser uses a category bigram language model. An implication of using N-grams (with discounting and backing-off strategies) is that recognition hypotheses may contain zero or multiple content words. The latter occur especially when lengthy OOV utterances are produced. In these cases, the first city name determines the value of the whole utterance and is passed on to the verification component; all other city names in the output are ignored.

2.6 Evaluation

The performance of our utterance verifier will be optimised and evaluated using the Confidence Error Rate (CER) [9]. This is the total number of false accepts (#FA) and false rejects (#FR) divided by the total number of all cases: correct (#COR) and incorrect (#INC). In order to find the corresponding points in the ROC curves, we also compute false accept rate (FAR = #FA/#INC) and false reject rate (FRR = #FR/#COR).

3 Implementation and experiments

In the introduction we argued that utterance verification is best served by an approach that acknowledges the diversity of errors. We stated that a one-dimensional confidence measure may not be optimal. This section first describes the way the distance measures are computed, and the experiments to compare the different measures. Next, the implementation of the CART procedure is explained, and the way in which these implementations are compared.

3.1 Path distance

Formalising our approach of how to measure the distance between two paths, we start by introducing some definitions. First, we assume that the phone alignment of both paths is known. In other words, for each feature vector we have the information about which HMM unit was aligned against it. Doing this by forced segmentation, we are also able to obtain the corresponding acoustic log-likelihood scores. The absolute difference between two scores for the same time frame is a log-likelihood ratio (LLR) score at frame level, as displayed by formula 1.

$$LLR(x_t|S_t^b, S_t^a) = LL(x_t|S_t^b) - LL(x_t|S_t^a) \tag{1}$$

where S_t^b and S_t^a are the states of the best and alternative hypotheses aligned against x_t, the feature vector at time t. LL(x|S) is the log likelihood score of vector x as computed with S's pdf. Next we combine the frame scores into a single score per content word. Formula 2 shows how we first take the average absolute LLR on phone level and next on word level.

$$\frac{1}{|W|} \sum_{\psi \in W} \left[\frac{1}{(\psi_e - \psi_s)} \sum_{t=\psi_s}^{\psi_e} abs\left(\, LLR(x_t|S_t^b, S_t^a) \, \right) \right] \tag{2}$$

where W is the content word we compute confidence for. $|W|$ denotes the number of phones in W. The index $\psi \in$ W runs over all phones of W with ψ_s and ψ_e being their respective start and end times.
In our experiments we used this formula to compute the distance between the recognised content word, i.e. the top candidate of the N-best list, and the runner up, if available. We refer to this distance as D_{rup}. At the same time, we compute the distance between the best content word and the optimal phone string resulting from free phone (FPh) recognition, which we call D_{fph} from now on.

3.2 The role of Language Models

One of the assumptions we made is that the distance between first best and best FPR path says something about the credibility of the acoustic score. At this point the question arises whether to use a phone language model (LM) in the FPR. Using an LM can help to minimise phone error rate, but one may wonder if this is a goal to aim for. By ignoring prior knowledge we will truly maximise acoustic likelihood. For the distance measure to the second most likely hypothesis (D_{rup}) however, the language model scores are important and should probably not be ignored. After all, if two candidates have equal acoustic scores, but one is enforced by the language model, the a-posteriori probability of an error will be minimised by selecting the hypothesis with the best score including the LM. Summarising, in the distance ratio D_{fph} the acoustic score ought to be sufficient, but in D_{rup} we should use both.

Experiment 1 In the second experiment we investigate the role of LMs in the computation of the likelihood scores. There are four combinations (D_{fph} with and without LM) x (D_{rup} with and without LM) and for each combination we computed the distance scores on a development and a test set. The distance between two paths is computed according to combination formula 2. The development set was used to train the coefficients of a linear combination function using LDA (for each of the four systems separately). With these functions we combined the distances of the test set and thus obtained a single confidence score for every utterance.

3.3 LC versus CART

In the first experiments we have only used an LC to combine the four confidence measures. Although LCs has the advantage of yielding a one-dimensional score, a linear separation may not be optimal in the face of multiple and unrelated causes of recognition errors. CART procedures have proven to be a very powerful alternative for LC in many speech recognition tasks. Therefore, we compare CART and LC for their power in distinguishing between correct and erroneous hypotheses at the output of our ASR system.
When training a CART, it is necessary to define an optimisation criterion for splitting a data set,

which is used to determine which split is 'best' for separating correctly from incorrectly recognised city names. In our situation we shift a threshold over each of the four numerical parameters to classify the ASR output. The threshold value that separates a maximally large yet 'pure' subset, is selected as the optimal split. In this sense, a pure set is a set with many members of one binary class (being 'correct' or 'incorrect' hypotheses), but just few of the other. Stated formally, we split on the threshold of either formula 3 or 4:

$$\hat{T} = \text{argmax}_T \left[\, 1 - FAR(T) - FRR(T)^{1/\psi} \, \right] \tag{3}$$

$$\hat{T} = \text{argmax}_T \left[\, 1 - FRR(T) - FAR(T)^{1/\psi} \, \right] \tag{4}$$

In words, these formulas express that the optimal threshold is at a value where false accept rate is minimal under the condition that false reject rate is 'very small' or vice versa. The strictness of very small is controlled with exponent ψ, with typical values around 2.0.

Experiment 2 We estimate the parameters of the tree on the same development set as used in Experiment 1. The evaluation is of course on the test set. Since the CART approach yields only one optimal separation scheme, there is no straightforward way to generate an ROC curve. Therefore, we compare the performance of CART and LDA in terms of Confidence Error Rate.

4 Results

All verification results are based on a single run of our recogniser on the test set. Since previous reports, like [6], we have improved performance and currently 14.4% of the utterances are incorrectly recognised.

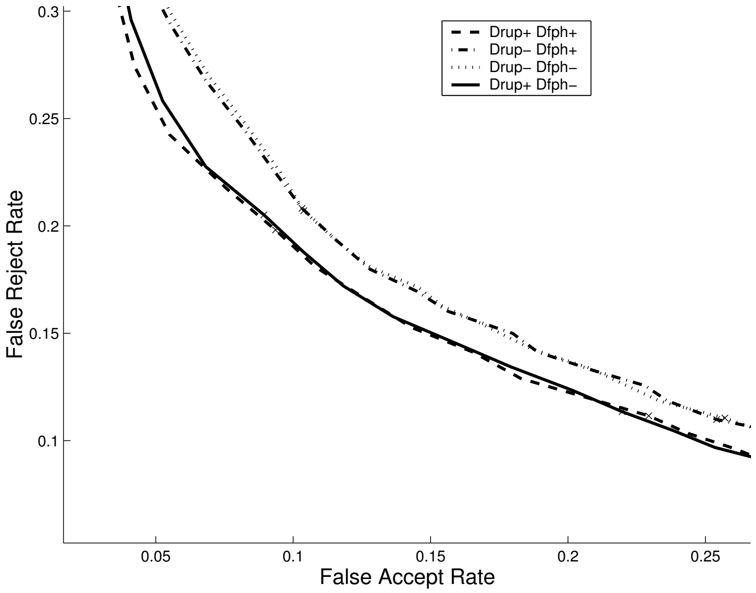

Fig. 3. ROC curves with and without language model contribution in the path distances

Figure 3 shows the ability to separate correct utterances from incorrect ones in terms of % False Accept and % False Reject. The +/- signs in the legend indicate whether LM scores are taken into account for the respective path distance. Table 1 shows the optimal CER values that correspond with these curves.

Table 1. CER, FAR and FRR when using the LM score (+lm) or disregarding it (-lm) in D_{rup} and D_{fph}.

D_{rup}	D_{fph}	%CER	%FAR	%FRR
-lm	+lm	11.2	51.3	4.4
-lm	-lm	11.4	53.0	4.4
+lm	+lm	10.5	54.2	3.1
+lm	-lm	10.5	54.1	3.2

The second experiment comprised building a CART. Its character cannot be illustrated by an ROC curve, because there is no threshold involved. Table 2 shows the test set CER evaluation for the tree that performed optimally on our development corpus.

Table 2. CER, FAR and FRR when verifying with a classification tree.

-	%CER	%FAR	%FRR
CART	9.4	31.3	5.6

5 Discussion

Starting to analyse the results of the first experiment, we see that all four curves have Equal Error Rates between 15% and 17.5% in Figure 3. So in absolute terms, they are all close to each other. When comparing the curves, we see that the two systems that take the language score distance into account for D_{rup}, have an ROC that is closer to the origin. This means that their ability to separate correct utterances from incorrect utterances is better. The CER values in rows 3 and 4 of Table 1 are significantly better than those in the first two rows (95% confidence). At the same time we see that the use of a language model to optimise the free phone recognition makes no significant difference. Here we see a confirmation of our idea that minimising phone error rate is not a goal to aim for. However, this is no evidence that omitting the LM is beneficial.

When comparing the two confidence measure combination methods, it appears immediately that the classification tree method is better than the linear combinations. The relative improvement of CART when compared with the best result obtained with LC is about 10%, which is significant. Although we did not perform a case-by-case analysis, this gives reason to believe that the problem of utterance verification is not optimally served by a one-dimensional confidence measure. Errors have diverse causes that can well be reflected in a multidimensional vector. The CART verifier has access to the vector components until the final decision moment.

We also examined the split parameters in our D-tree. It appeared that D_{rup} was in fact the most informative variable for the population at the root node. The next split, down either of the branches, was based on D_{fph}. The total number of nodes in the tree amounts to 8 and all four cues were used.

6 Conclusion

In this paper we compute confidence scores on the basis of mutual distances between various Viterbi paths. The path of the best recognition hypothesis is compared with the best path of a Free Phoneme Recogniser and with the path of the runner-up candidate in the N-best list. In two experiments we tested several linear combinations of the two distances and two additional cues. Studying the ROC curves, we conclude that the distance measure between the first best and second best candidate should take the language model score into account. Due to the nature of speech recognition errors and the way their occurrence is reflected in our four different cues, we found optimal performance when the cues are being kept separated instead of mapping them to a single confidence measure. With a classification and regression tree (CART) we measured a relative confidence error rate improvement of 10%.

Finally we would like to point out that there are additional cues for confidence that we did not consider in this paper, although we did report on their potential contribution in the work described in [6]. The number of syllables (or the amount of speech) of the recognised word appeared to be especially powerful. Decisions based on more acoustic data are generally more reliable, and the probability of correct classification is higher. Moreover, it may well be that prosodic (and other) parameters that are not particularly effective in a linear separation can contribute valuable information in specific parts of the complex space. We plan to investigate this issue in the near future.

References

1. L. Boves, D. Jouvet, J. Sienel, R. de Mori, F. Béchet, L. Fissore, P. Laface ASR for Automatic Directory Assistance: the SMADA Project Proc. of ASR2000 Paris (2000), pp unavailable
2. D. Charlet, G. Mercier, G. Jouvet: On Combining Confidence Measures for Improved Rejection of Incorrect Data. Proc. of Eurospeech '01. Aalborg (2001), pp. 2113–2116
3. S. Kamppari, T. Hazen: Word and Phone Level Acoustic Confidence Scoring. Proc. of ICASSP '00, vol III. Istanbul (2000), pp. 1799–1802
4. T. Hazen, I. Bazzi: A Comparison and Combination of Methods for OOV Detection and Word Confidence Scoring. Proc. of ICASSP '01, vol I. Salt Lake City (2001), pp. 397–400
5. B. Tan, Y. Gu, T. Thomas: Word Level Confidence Measures Using N-best Sub-hypotheses Likelihood Ratio. Proc. of Eurospeech '01. Aalborg (2001), pp. 2565–2568
6. G. Bouwman, L. Boves: Using Information on Lexical Stress for Utterance Verification. Proc. of ITRW on Prosody in ASRU. Red Bank (2001), pp. 29–34
7. L. Breiman(ed) et al.: Classification and Regression Trees. Chapman & Hall. 1998
8. J. Sturm, H. Kamperman, L. Boves, E. den Os: Impact of Speaking Style and Speaking Task on Acoustic Models Proc. of ICSLP '00, vol I. Beijing (2000), pp. 361–364
9. F. Wessel, K. Macherey, R. Schlüter: Using Word Probabilities as Confidence Measures. Proc. of ICASSP '98, vol I. Seattle (1998), pp. 225–228