PHONE-DURATION-BASED CONFIDENCE MEASURES FOR EMBEDDED APPLICATIONS

Silke Goronzy, Krzysztof Marasek, Ralf Kompe, Andreas Haag

Sony International (Europe) GmbH, Advanced Technology Center Stuttgart (ATCS)
Home Network Company Europe, Man Machine Interfaces
Hedelfinger Str. 61, D-70327 Stuttgart, Germany
Fon: +49-711-5858-456, Fax: +49-711-5858-199, {goronzy, marasek, kompe, haag}@sony.de

ABSTRACT

In order to detect misrecognitions that may result from a mismatch between training and testing data, we use a confidence measure (CM) that collects a set of features during recognition and from the N-best list that is output by the recognizer. A neural network (NN) then calculates the probability that the utterance was recognized correctly based on these features. Since for misrecognized utterances the resulting phoneme alignments are often erroneous, we introduced some new features that are based on phoneme durations. The durations found by the recognizer are compared to the durations present in the training data base and the results of these comparisons serve as input for the NN. A great advantage of the duration-related features is that they are independent of the recognizer in contrast to e.g. acoustic score-based features. We also use some score-related features that have proven to be useful in the past. Simultaneously with determining the confidence for a recognition result, we try to detect if in case of a misrecognition the utterance was an out of vocabulary (OOV) utterance. Using the complete set of 46 features we can achieve a correct classification rate of 90%. The word error rate can be reduced by 92% at a false rejection rate of 5.1% on a test task that consists of 35 speakers and includes more than 50% OOV utterances. OOV words were detected correctly in 91% of the cases. The presented CM is also used in a semi-supervised speaker adaptation scheme.

1. INTRODUCTION

In speech recognition systems the problem arises that there often is a severe mismatch between training and testing conditions, like e.g. in car environments, where it is impossible to cover all possible testing conditions in the training. This may cause misrecognitions in the later application. Since the cost of processing a misrecognized command may be high, it is often more desirable to re-prompt the user rather than to proceed with the wrong command. In order to increase the reliability of the system, confidence measures can be used to decide if the probability that an utterance was misrecognized is high and in that case to reject it and re-prompt the user. There are many approaches to deal with this problem. One is to collect a set of features during the search and then combine these to formulate the final CM. Several studies tested the combination of a set of features and compared this to the performance of each feature alone and found that combining them outperforms either feature if taken alone, cf. [1, 2].

The CM applied in our isolated word, command&control approach uses such features, that are not used explicitly during recognition. We used two groups of features, the newly introduced ones being the ones measuring phone duration distributions. This is motivated by the observation that in case of misrecognitions often a severe mismatch between the segmentation found by the recognizer (and thus the phoneme durations) and the durations present in the training data can be found. We also include features measuring the spectral homogeneity within a recognized word, because in case of misrecognitions often a segment covers several 'real' phone segments with different spectral properties. This often results in sequences of very long followed by very short phonemes and vice versa. The second group of features employs some well known features that are related to the acoustic score and try to model the uncertainty during the Viterbi search. The features we used were frame-, phone- and word level-based, respectively. As the classifier we use a NN[1] which computes the probability for correct/incorrect recognition. In additional experiments we try to judge whether a rejected utterance was an OOV word or a misrecognized one.

The use of CMs is useful also for other purposes, e.g. to guide unsupervised speaker adaptation. Only words that are classified as being correctly recognized are used for adapting the acoustic models. We compare the performance of unsupervised adaptation to adaptation which is guided by our

[1] We used SNNS, which was developed by the University of Stuttgart

CM. We call this semi-supervised adaptation.

The simple topology of the NN and the straightforward computation of the confidence features allows the implementation of the system on an embedded platform.

2. CM FEATURES

2.1. Phone-Duration-Based Features

We used 13 features that are related to phoneme durations and speaking rate. For the training of the NN the distributions of durations of all phones are determined, based on a forced alignment of the training data. The durations are additionally smoothed and are later compared to the phoneme durations that were determined by the alignment of the recognizer during testing. Since it is well known that the speaking rate strongly influences the phoneme durations, we estimated the speaking rate and normalized the found durations accordingly. The speaking rate was estimated as follows, cf. [3]:

$$\alpha = \frac{1}{N} \sum_{i=1}^{N} \frac{d_i}{\bar{x}_p} \,, \tag{1}$$

where *N* denotes the number of phones in the observed utterance and/or in the past few utterances, d_i is the duration of the *i-th* phone segment (recognized as phone *p*) in the utterance and $\bar{x}_p$ is the mean length of the corresponding phone *p* learned during training.

Some of the features were multiply used, e.g. not normalized, normalized by the number of frames, by the acoustic score of the best hypothesis or by the speaking rate. The feature set comprises the following features:

1. *n_toolong01*, *n_toolong05*: Number of phones in the best hypothesis that are longer than the 0.01 and 0.05 percentile, respectively, compared to the training data (correspondingly we used the same features for too short durations)

2. *sequence*: Number of sequences of phone pairs within one utterance where the first phoneme was too long and the second one too short (or vice versa) (using the 0.01 percentiles)

3. speaking rate-related features: Current and average (of the last n utterances) speaking rate and its standard deviation and absolute difference between the current and average speaking rate.

2.2. Additional Features

In addition to the duration-based features described above we used 31 features that are related to the acoustic score. These compare in various ways the acoustic scores of the N-best list at frame-, phoneme- and word-level. A complete list of the features used can be found in [4]. These features are known to depend strongly on the recognizer, i.e. the front-end etc.

	# patterns	corr	wrong	OOV
train	37718	18859	3022	15837
eval	250	1795	330	2125
test	16186	7071	1022	8093

Table 1: Number of patterns used for training, evaluation and testing the NN

3. CLASSIFICATION

The NN was used to classify the recognition result in a post processing step. The utterance was recognized as usual and based on the extracted features a decision whether to accept or reject the utterance was made by the NN. We trained a feed forward net, that consists of one hidden layer only. We used 46 input nodes, 8 hidden nodes and 2 or 3 output nodes, respectively. The training data for the NN comprised clean speech only and also included a large amount of OOV words. The detailed statistics concerning the patterns used for training, evaluation and testing can be found in Table 1. One pattern corresponds to the features extracted from one utterance. The data for training the NN was obtained using our standard recognizer, that will be described in more detail in section 4.1. Since we use some features related to the acoustic score, we divided the training set into into two parts. One was used for training the monophone models, the second set was used to generate the training patterns for the NN, using the previously generated monophones. This was necessary to avoid a possible influence of the training data on the acoustic score if the same data would be used for training the models and determining the reference durations. The NN training data was automatically labelled as being correctly recognized by the speaker independent (SI) system or not. The target output for the 2 output nodes of the NN were either '1 0' (recognized correctly) or '0 1' (misrecognized). In a second set of experiments we used a NN with 3 output nodes. The first two output nodes have the same meaning as before, the third output node is to indicate, whether a misrecognized word was an OOV word or not ('1' means OOV). So possible outputs are '0 1 1' or '0 1 0' in case of a misrecognition. For correctly recognized words only '1 0 0' is possible. During testing the NN outputs values between 0 and 1. The final decision threshold is then simply 0.5. This means that if the first output is greater than 0.5 and the second is smaller than 0.5, the utterance is considered as being correctly recognized. If both values were greater or lower than this threshold, the recognition results could not be classified. This happened in 0.3% of the cases. Correspondingly if the third output node was greater than 0.5, the utterance was marked as being OOV.

	CER	C_a	F_a	C_r	F_r
baseline (SI)	-	43.7	56.3	-	-
2 out	9.67	38.45	4.62	51.55	5.05
3 out	10.29	37.9	4.62	51.62	5.67

Table 2: Classification error rates (%) of a NN with 46 input nodes and 2 and 3 output nodes, respectively

4. EXPERIMENTS AND RESULTS

4.1. Experimental Setup

Training was conducted using a German isolated word data base recorded in our sound treated studio. It consists mainly of isolated words and short phrases. The vocabulary size was 375. The speech was sampled at 16 kHz and coded into 25 ms frames with a frame shift of 10 ms. Each speech frame was represented by a 38-component vector consisting of 12 MFCC coefficients (no energy) and their first and second time derivatives. The first and second time derivatives of the energy are also included. We trained 3-state, monophone HMM models with 1 Gaussian per state using 34281 utterances from 80 speakers. The choice of such a simple model was due to the memory and speed requirements of the target platform, an embedded system for a command&control application. The corpora we used for testing were a German address corpus with approximately 23 utterances per speaker and a command&control corpus with approximately 234 utterances per speaker, so around 260 utterances per speaker in total. We then added the same number of OOV utterances for each speaker, resulting in 540 utterances per speaker. The test set consisted of 35 speakers.

4.2. NN results

We first tested the full set of features using a net with 2 and 3 output nodes, respectively. We achieved the results that are listed in Table 2.

The performance of our CM was measured in terms of classification error rate (CER), which is the number of misclassified patterns divided by the total number of patterns. Also we listed the correct and false alarm rates (C_a and F_a, respectively) and correct and false rejection rates (Cr and Fr, respectively). The C_a rate directly corresponds to the recognition rate of the SI system. The very poor baseline performance of 43.7% can be explained by the fact that we included 50% OOV words in the test set to assess the capability of the NN to distinguish between misrecognized and OOV utterances. False alarms are those utterances that have been classified as being correctly recognized although they were misrecognized, so this corresponds to the word error rate (WER) of the SI system. Both NNs correctly reject more than 50% of the utterances (C_r) although at the cost of falsely rejecting 5.1% and 5.7% of the correctly recognized ones (F_r), respectively. Table 2 shows that rejecting all utterances that were classified by the NN as being misrecognized reduces the F_a-rate (WER) of the SI system by more than 90% (from 56.3% to 4.6%). However, this is at the cost of also rejecting 5.1% of the correctly recognized utterances. The results for the NN with 3 output nodes are slightly worse. However, it provides us with further information, such as if the word was an OOV word or not. This information is shown in Table 3. In 88.6% of the C_r-cases the NN classified OOV words correctly. Especially for dialogue systems it could be beneficial for the course of the dialogue, to know if the utterance was an OOV word or simply misrecognized. This knowledge can greatly influence the following dialogue steps. In a third set of experiments we tried to test our new duration-based features against the well-known acoustic score-related ones. We trained new NNs using 13 purely duration- and speaking rate-related features, 31 purely score-related features and 35 score- and speaking rate-related features. Table 4 summarizes the results we obtained. The set that uses only score-based features performs best, followed by the full feature set and the set using the score- and speaking rate-based features. The duration-based features achieve the worst, however still acceptable, results. Surprisingly combining the duration- and score-based features does not outperform either set if taken alone. Still we consider the duration-based features as valuable, since they are completely independent of the recognizer. This is especially important if the NN-based CM is to be used in different embedded applications. Using the score-related features would require a new NN training for each application and each platform, since such parameters as e.g. front-end strongly influence the acoustic scores that are later produced during search.

	corr	wrong
3 out	88.6	11.4

Table 3: OOV classification results (%)for the NN with 3 outputs, determined on the C_r-cases only

	CER	C_a	F_a	C_r	F_r
baseline (SI)	-	43.7	56.3	-	-
ac feat only	9.03	39.48	5.15	50.8	3.88
ac + spk rate	8.95	37.39	3.73	51.19	5.21
dur feat only	16.4	36.09	8.85	47.47	7.57
all	10.79	37.5	4.6	51.72	6.19

Table 4: Classification results (%) for different feature sets

4.3. Semi-supervised Adaptation

For most command&control applications it is not feasible to collect enough speech data from one speaker to train speaker

#utterances	400	600	1000	2000
unsupervised	42.9	38.7	39.6	42.5
semi-sup. our CM	39.6	37.7	37.7	37.3
semi-sup. perf. CM	46.2	37.7	40.6	44.8

Table 5: Improvements in % WER w.r.t. the SI system

dependent (SD) models. However if the devices are used for a longer time by the same person, some kind of speaker adaptation should be employed to improve the performance of the SI system. Supervised adaptation schemes, that need a relatively large amount of adaptation data and where the user has to read a predefined text are also not desired. We need an unsupervised approach that allows the user to start using the system right away. But it should have the capability to adapt to the user's voice while he is actually using the system and is not aware that some adaptation is going on. The weakness of such unsupervised adaptation schemes is, that they often fail if the baseline performance of the recognizer is too low and too many misrecognitions occur. Since the adaptation has to rely on the recognizer output to be the spoken word, misrecognitions cause the wrong models to be adapted. At this point we integrate our CM into the adaptation process. We no longer use every utterance, no matter if it was misrecognized or not, for adaptation as we did in the unsupervised approach, but we use only those utterances that were accepted by the CM. For adaptation we used a combination of MLLR and MAP adaptation. For MLLR one global regression class was used, thus achieving a fast but coarse channel and speaker adaptation. This was done for the fist 15 utterances. From then on MAP was used to achieve a finer adaptation using the MLLR-adapted models as prior information, cf. [5, 6]. The front-end was the same as described above, but another test set was used. We used 6 speakers for testing and 2000 utterances, exclusively comprising commands and no OOVs, for each speaker in total. We tested after a different number of utterances. The improvements w.r.t. the SI system averaged over the 6 test speakers are shown in Table 5.

For clean speech with an initial error rate of 22.8% the unsupervised approach reduces the WER of the SI system. The improvements get bigger the more data becomes available. The semi-supervised approach cannot improve these results any further, the results are even slightly worse. Even using a 'perfect CM' for supervision, in which we used only correctly recognized utterances for adaptation does only slightly improve the results for a large number of utterances. The reason could be that the initial error rates are relatively low and that the effect of not using these few utterances for adaption is nullified by the reduced number of utterances when rejecting some in the semi-supervised approach. On the hand, this demonstrates the robustness of our adaptation approach.

5. CONCLUSION

We presented new features for a CM approach, that uses a NN as a classifier. These features are based on phoneme duration statistics, that were obtained from the training data. Together with features that are related to the acoustic score present in the N-best output of the recognizer we achieved a CER of 9.7% at a false rejection rate of 5.1%. Simultaneously we succeeded in identifying OOV words in 91% of the C_r-cases. Using the duration-based features alone we achieve a CER of 16.4%. Although being worse than the score-based feature set, these features are independent of the recognizer and a retraining of the NN is not necessary if e.g. the front-end is changed, while using score-related features would require such a retraining. The features we used are mostly related to single words. However, this approach can be easily extended to LVCSR systems.
We combined the CM using the full feature set with speaker adaptation, such that adaptation is conducted in a semi-supervised manner and only utterances that were accepted by the CM were used for adaptation. The WER reduction of 42.5% achieved by the unsupervised approach could not be improved any further. Even using a perfect CM did only yield slight improvements for large amounts of adaptation data. This demonstrates the robustness of our unsupervised adaptation approach.

6. REFERENCES

[1] Wendemuth, Rose, and Dolfing. Advances In Confidence Measures For Large Vocabulary. In *1997 International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 705–708. IEEE, 1997.

[2] Kemp and Schaaf. Estimating Confidence Using Word Lattices. In *5th European Conference on Speech Communication and Technology*, volume 2, pages 827–830. European Speech Communication Association (ESCA), 1997.

[3] Ralf Kompe. *Prosody in Speech Understanding Systems*. Springer Verlag, 1997.

[4] Goronzy, Marasek, Kompe, and Haag. Phone-Duration-Based Confidence Measures for Embedded Applications. In *International Conference on Spoken Language Processing*, volume 4, pages 500–503, 2000.

[5] Goronzy and Kompe. A MAP-like weighting scheme for MLLR speaker adaptation. In *6th European Conference on Speech Communication and Technology*, volume 1, pages 5–8. European Speech Communication Association (ESCA), 1999.

[6] Goronzy and Kompe. A Combined MAP + MLLR approach for speaker adaptation. In *Proceedings of the Sony Research Forum 99*, volume 1, pages 9–14, 1999.