Radboud Repository

Radboud University Nijmegen

PDF hosted at the Radboud Repository of the Radboud University Nijmegen

The following full text is an author's version which may differ from the publisher's version.

For additional information about this publication click this link.
http://hdl.handle.net/2066/75070

ACCUMULATED KULLBACK DIVERGENCE FOR ANALYSIS OF ASR PERFORMANCE IN THE PRESENCE OF NOISE

Febe de Wet, Johan de Veth, Bert Cranen, and Louis Boves

A^{2}RT, Department of Language and Speech
University of Nijmegen, The Netherlands
{f.dewet, j.deveth, b.cranen, l.boves}@let.kun.nl

ABSTRACT

In this paper, the accumulated Kullback divergence (AKD) is used to analyze ASR performance deterioration due to the presence of background noise. The AKD represents a distance between the feature value distribution observed during training and the distribution of the observations in the noisy test condition for each individual feature vector component. In our experiments the AKD summed over all feature vector components shows a high correlation with word error rate and AKD computed per component can be used to pinpoint those feature vector components that substantially contribute to recognition errors. It is argued that the distance measure could be a useful evaluation tool for analyzing the strengths and weaknesses of existing noise robustness approaches and might help to suggest research strategies that focus on those elements of the acoustic feature vector that are most severely affected by the noise.

1. INTRODUCTION

Usually ASR engines are trained with speech that has been acquired in a relatively quiet environment. Thus, the statistics of the individual components of the acoustic vectors mainly reflect variation that can be attributed to intra- and inter-speaker differences in the speech sounds. In the presence of background noise, some or all of the acoustic vector components will show statistics that differ from those on which the ASR engine was trained. As a consequence, 'noisy' acoustic vectors associated with a given speech unit may be differently distributed compared to the probability density function (pdf) that describes the clean data for that unit in model space. Such differences will likely increase word error rate (WER).

At least two ways exist to make ASR more noise robust: (1) Find a feature representation that is inherently noise robust, i.e., insensitive to background noise in the sense that the observed feature values are hardly affected by the presence of noise, and (2) apply noise reduction, i.e., estimate disturbances caused by the background noise and compensate for these disturbances.

The effectiveness of a given noise robustness approach is conventionally evaluated by monitoring WER. However, WER is a crude measure, that does not disclose the mechanisms underlying some improvement (or the causes of a failure to find improve-

ment). A tool that does provide more direct access to the underlying mechanisms is therefore needed. For the case of inherently noise robust features, such a tool should provide a metric with which the change in observation distributions in acoustic feature space due to the noise can be quantified (and subsequently be minimized). The same holds for noise reduction techniques: if we had a tool to measure the distribution differences between clean data and noise reduced data, it would be easier to design the 'ideal' noise reduction technique.

From the literature on noise robust ASR, it is evident that the relation between WER and signal-to-noise ratio (SNR) is far from simple. At a given SNR, the error rate is strongly dependent on the type of noise and the type of acoustic features. Until now it has not been possible to predict which type of feature representation is most resistant to a particular type of noise. Here too, a tool that allows one to analyze the distance between clean training data and noisy test data would be a step towards a better understanding of the issue.

In this paper, we present a measure based on the Kullback divergence [1, 2] as a means to describe training-test mismatch. The measure describes the average distance between the statistical distributions of the test data and the distributions as observed on the set of train data. An important property of this measure is that it allows to quantify the relative contributions of individual components of the acoustic vectors. As a result, it is possible to identify those vector components that contribute most to the distance measure. We therefore think that the distance measure may be a first step towards the desired tool referred to earlier.

We want to illustrate the viability of this approach in the context of a digit recognizer that has been trained on clean data and tested in (simulated) noisy conditions. To that aim, we investigated the distance measure in combination with the changes in WER when training-test mismatch is selectively and artificially removed from those vector components that appear to have the largest relative distance contributions. This allowed us to study whether repairing components with a large contribution to the distance substantially increases recognition performance, and to confirm whether the measure has indeed the intended diagnostic properties.

2. ACCUMULATED KULLBACK DIVERGENCE

The Kullback divergence is a well-known measure for the distance between two statistical distributions [1, 2]. If we denote the observation distributions for the train and test condition as d_{trn} and d_{tst}, respectively, the Kullback divergence $\mathcal{K}$ for quantifying training-

test mismatch is defined as

$$\mathcal{K}(d_{trn}, d_{tst}) = \int [d_{trn}(\mathbf{x}) - d_{tst}(\mathbf{x})] \log \frac{d_{trn}(\mathbf{x})}{d_{tst}(\mathbf{x})} d\mathbf{x}, \quad (1)$$

where $\mathbf{x}$ denotes the observation vector. $\mathcal{K}$ is a symmetric distance measure, and is equal to zero when the two distributions d_{trn} and d_{tst} are identical. Under the assumption that the observation vector components $x_k, k = 1, \ldots, K$ (with K the dimension of the observation vector) are independent, $\mathcal{K}(d_{trn}, d_{tst})$ can be computed as the sum of the Kullback divergence for each component $\mathcal{K}_k(d_{trn}, d_{tst})$

$$\mathcal{K}(d_{trn}, d_{tst}) = \sum_{k=1}^{K} \mathcal{K}_k(d_{trn}, d_{tst}), \quad (2)$$

with

$$\mathcal{K}_k(d_{trn}, d_{tst}) = \int [d_{trn}(x_k) - d_{tst}(x_k)] \log \frac{d_{trn}(x_k)}{d_{tst}(x_k)} dx_k. \quad (3)$$

In this paper we modeled d_{trn} in terms of the Gaussians that are used to model the train set. Assuming one uses I states for each HMM l and M Gaussian pdfs to model each state i, the distribution of the train data d_{trn} for state i of HMM l is described as

$$d_{trn}(x_k, l, i) = \sum_{m=1}^{M} w_m G(x_k, l, i, m), \quad (4)$$

where $G(x_k, l, i, m)$ denotes the mth weighted mixture component (with weight w_m) in the Gaussian mixture pdf for component k of the observation space for the ith state in HMM l.

For the computation of d_{tst}, a segmentation is needed for each utterance in the test set, so that the association between each observation vector of each utterance and an HMM state is known. Once the segmentation is known, the histogram of test observations $H(x_k, l, i)$ compiled for each observation component k of state i in model l is used to compute $d_{tst}(x_k, l, i)$ as follows:

$$d_{tst}(x_k, l, i) = \frac{1}{N(l, i)} H(x_k, l, i), \quad (5)$$

where $N(l, i)$ denotes the total number of observation vectors associated with state i of model l.

Using Eqs (3), (4), and (5), the accumulated Kullback divergence (AKD) per feature component $AKD(k)$ is computed as:

$$AKD(k) = \sum_{l=1}^{L} \sum_{i=1}^{I} \mathcal{K}_k \left(\sum_{m=1}^{M} w_m G(x_k, l, i, m), \frac{H(x_k, l, i)}{N(l, i)} \right). \quad (6)$$

So, by studying $AKD(k)$ for $k = 1, \ldots, K$, one should be able to tell whether some observation vector components contribute more to the *overall AKD* (i.e., $\sum_{k=1}^{K} AKD(k)$) than others.

3. EXPERIMENTAL SET-UP

3.1. Clean speech material

The speech material for our experiments was taken from the Dutch POLYPHONE corpus [3]. Speech was recorded over the public switched telephone network in the Netherlands, using a primary rate ISDN interface and a sampling frequency of 8 kHz. The POLYPHONE corpus contains various examples of (read) speech utterances. Only the connected digit items were used in our current investigation. The number of digits in each string varied between 3 and 16. A set of 1,997 strings (16,582 digits) was used for training. Care was taken to balance the training material with respect to gender, region (an equal number of speakers from each of the 12 provinces in the Netherlands) and the number of tokens per digit. 504 digit string utterances (4,300 digits) were used for cross-validation during training. An independent test set of 1,008 utterances (8,300 digits) was used for evaluation. The cross-validation and independent test sets were balanced according to the same criteria as the training material.

3.2. 'Noisified' speech material

Recognition performance was evaluated under two different simulations of adverse acoustic conditions. Babble and factory noise from the Noisex CD were chosen as the noise conditions for the current experiments. For all practical purposes, the babble noise may be considered as stationary. The factory noise contains a number of hammer blows and could therefore be considered as an example of non-stationary noise. In terms of their long time average spectra, both babble and factory noise can be classified as relatively broad-band noise. The Noisex signals contain broad-band frequency information while the information content of the signals in our database is limited to the frequency range of the public switched telephone network in the Netherlands. As an approximation of the channels frequency response, the Noisex signals were band-pass filtered before they were added to the clean signals. The addition was performed such that the SNR level of the resulting signals was 10 dBA.

3.3. Acoustic pre-processing

A pre-emphasis factor of 0.98 and a 25ms Hamming window shifted with 10ms steps were used to prepare the data for spectral analysis. A 256 point FFT was subsequently calculated for each windowed segment. From these spectra, 16 mel-scaled log-energy values were calculated. The filters in the mel bank were triangularly shaped, half overlapping and uniformly distributed on a mel-frequency scale between 122 and 2146 mel, corresponding to 80-4000 Hz on a linear frequency scale. 12 MFCCs were derived from the log of the mel bank outputs using the Discrete Cosine Transform. Cepstral mean subtraction (CMS) was applied as a channel normalization technique. We used the off-line version of the CMS algorithm, i.e. the cepstral mean was calculated per utterance. The first derivatives of the MFCCs were also computed and added to the vector of 12 channel normalized feature values. The HTK normalized log-energy (LogE) and delta LogE values of each frame were also included in the acoustic feature vectors [4].

3.4. Hidden Markov Modelling

Continuous density hidden Markov models (HMMs) were used to describe the statistics of the speech sounds. The ten Dutch digit words were described in terms of 18 *phone* models. Two additional models were used to represent the statistical properties of the silence and background noise (non-speech) in the recordings of the POLYPHONE database. Each phone unit was represented as a left-to-right HMM of three states. Only self-loops and transitions to the next state were allowed. All HMMs were implemented using diagonal covariance matrices and 16 Gaussian

mixtures components per state. HTK was used for training (with cross-validation) and testing [4]. The recognition syntax used during cross-validation and testing allowed for digit strings varying in length from 3 to 16 digits to be recognized, without prior knowledge of the length of a particular string. The syntax also allowed silence and noise to be recognized between consecutive digits as well as at the beginning and the end of each utterance.

3.5. Experiments

We first did a series of recognition experiments to establish baseline performance for the clean, matched condition, and for the conditions where 10 dBA babble noise and factory noise were added to the speech signals, respectively. Using the HMM state segmentations obtained from the Viterbi decoding of each test utterance, the AKD was computed for each feature vector component. Next, the AKD per feature vector component for the clean, matched condition was compared to the corresponding AKD per component for each of the mismatched conditions. The aim of this comparison was to identify those feature vector components that are most affected by the additive noise in terms of AKD.

In a second set of experiments, we replaced the noisy tracks of the most affected feature vector components with the feature tracks observed in the clean, matched condition. Then, we measured recognition performance again and also determined AKD for each feature vector component. The aim of these experiments was to determine whether feature vector components marked as most affected by the noise (as measured in terms of AKD) are indeed important for the reduced recognition performance observed in the presence of additive noise. As an additional check, we replaced the noisy track of a feature vector component that was much less affected according to its AKD value, and determined recognition performance and AKD after replacement.

4. RESULTS

The WER and overall AKD values determined for the clean, matched condition, and for the conditions where babble and factory noise were added to the clean test speech signals are shown in Table 1. WER was measured as $\frac{S+D+I}{N} \times 100\%$, where N is the total number of words in the test set, S denotes the total number of substitution errors, D the total number of deletion errors, and I the total number of insertion errors. The results in Table 1 indicate that the recognition performance decreases significantly in the presence of additive noise and that this decrease is accompanied with an increase in overall AKD.

Table 1. *Baseline recognition performance and overall AKD for the clean condition, 10 dBA babble, and 10 dBA factory noise.*

acoustic condition	WER	overall AKD
clean	3.6	267
babble	28.4	654
factory	31.6	769

Next, the AKD was analyzed as a function of the feature vector component for the three baseline conditions. The results are shown in Fig. 1A. The vertical dashed line in this Figure indicates the boundary between the static and delta coefficients. It can be seen that the largest AKD values for the clean condition are

found for the first few cepstral coefficients, LogE, and their corresponding deltas. Furthermore, all of the AKD values measured in noise exceed those measured in the clean condition for all feature components. Moreover, the increase of AKD values appears to be higher for some components than for others. The components that are mostly affected are LogE, $c_1, \ldots, c_4$, and their corresponding deltas. The two components that are most severely affected are LogE and c_1, irrespective of the noise type. For both types of noise, the higher cepstral coefficients and their corresponding deltas appear to be least affected. Finally, these results suggest that (with a few exceptions) the degree to which a cepstral component is affected is inversely proportional to its index, both in clean and in noisy conditions.
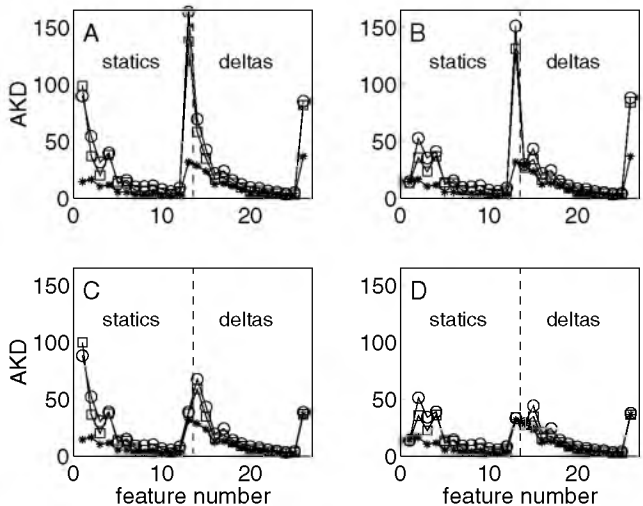

Fig. 1. Kullback divergence as a function of feature vector component, for the clean condition (∗), babble noise (□), and factory noise (○). Components 1 thru 12 correspond to $c_1, \ldots, c_{12}$, 13 is LogE with 14 thru 26 the corresponding delta-coefficients. A: Original data. B: c_1 matched. C: LogE matched. D: Both c_1 and LogE matched.

In the second set of experiments, we replaced the feature tracks of the static and delta coefficients for the two components that showed the largest AKD contribution. We replaced the feature tracks in the noisy test data by the corresponding tracks from the clean data for c_1 alone, for LogE alone, and for both these coefficients simultaneously. Then, we determined the WER and the AKD per component in these three conditions for each of the two noise types. The recognition results are shown in Table 2, together with the overall AKD values; the AKD results per component are shown in Figs 1B-D. In addition to these three experiments, we also replaced the noisy track of c_7 alone. According to Fig. 1A, c_7 is one of the components with a small contribution to overall AKD. The WER and overall AKD results for this experiment are shown in the last row of Table 2.

As can be seen in columns two and three of Table 2, recognition performance is substantially improved when either c_1 or LogE alone is replaced. Moreover, replacing both noisy components by the corresponding clean tracks reduces the WER even further. Figs 1B-D show that the AKD values of replaced components are lowered to values close to those observed in the clean, matched

3

condition. The results in Figs 1B-D also indicate that replacement of one feature track hardly changes the AKD contribution of other feature components.

Table 2. *Recognition performance and overall AKD for the baseline, and for babble and factory noise when either mismatched c_1, mismatched LogE, both mismatched c_1 and LogE, or mismatched c_7 are replaced by the clean, matched values.*

	WER		overall AKD	
noise	babble	factory	babble	factory
baseline	28.4	31.6	654	769
c_1	19.5	22.1	528	643
LogE	18.5	20.5	498	573
c_1 & LogE	12.3	13.3	376	458
c_7	27.1	29.7	649	757

In Table 1, it can be seen that overall AKD increases due to the presence of noise. Table 2 shows that overall AKD (just like WER) is reduced when a mismatched feature component is replaced by its corresponding clean track. Moreover, both overall AKD and WER are reduced more when more components are artificially repaired. Finally, both overall AKD and WER only decrease slightly when the replacement is done for c_7, which has a small contribution to overall AKD. In order to determine the relation between overall AKD and WER, we computed the correlation coefficient and found ρ equal to 0.98.

5. DISCUSSION AND CONCLUSIONS

In this paper we studied recognition performance for automatic speech recognition in the context of training-test mismatch due to additive noise. More in particular, we studied the accumulated Kullback divergence (AKD). This measure constitutes a distance between the feature value distribution observed during training and the distribution of the observations in the noisy test condition for each individual feature vector component. We considered a rather limited amount of different recognition conditions, i.e. only two different noise types (Noisex babble and factory noise, artificially added to the clean test speech signals at one SNR=10 dBA), and one single feature representation (MFCCs with LogE, and the corresponding deltas). Moreover, the impact of the additive noise on the acoustic feature vector was reduced in an artificial way.

Within the limits of the current experimental set-up, we found that artificially repairing a component with a high AKD contribution reduced both the overall AKD and the proportion of ASR errors at the word level. Therefore, we think that the AKD per component can identify those feature vector components that have the largest impact on WER.

While overall AKD is computed at the HMM state level and WER is determined at the word level, both measures are based on the output of the same Viterbi recognition process. This makes them closely related. In fact, we found a high correlation ($\rho = 0.98$) between the overall AKD and WER. This suggests that overall AKD can be safely interpreted as an alternative way to evaluate recognition performance.

It is tempting to speculate that the AKD per component can be used for different diagnostic purposes. In this paper we saw examples showing that the contribution to the overall AKD can be quite different for different feature vector components, not only in situations with training-test mismatch, but also in a clean, matched condition. In the case of training-test mismatch, this difference can provide valuable information about which components suffer most from the noise. In the clean matched condition, it tells how well each individual component has been modeled. According to Fig. 1, LogE and its corresponding delta appear to have the largest AKD in the clean, matched condition. This suggests that there is still room to improve the LogE estimate for the baseline ASR system used in this paper. As a second possible application, we mention the study of feature types that are presumed to be inherently robust to the presence of noise. As an example, it would be interesting to study the AKD per component for formant-like features. One would hope that AKD per component observed in a noisy condition for formant-like features would hardly differ from the AKD per component in the clean condition. Should this prove to be true, one could consider using such formant-like features together with more traditional feature vector components like MFCCs to improve noise robustness.

Another possible application could be the evaluation of the effectiveness of noise reduction techniques. We expect that proven noise reduction techniques (e.g., [5]) succeed in reducing overall AKD. As we have seen in this study, some feature components are more affected in terms of AKD then others. By focussing on the reduction of those AKD contributions that are still most affected after noise reduction, one could hope to obtain clues that help to optimize such techniques even further. Experiments are under way to evaluate AKD per component for features computed after application of noise reduction.

The way in which feature values are distributed in the presence of noise is determined by the combination of the type of noise and the type of acoustic pre-processing steps. Study of the AKD is one way to evaluate the impact on recognition performance of the combination of noise type and feature representation on a per component basis. Of course, the AKD per component cannot tell how to improve the feature representation. Nevertheless, we believe that knowledge about which feature components are most affected in terms of AKD can help to make well-reasoned choices when designing an acoustic front-end for noise robust ASR. Clearly, additional experiments are required to verify whether the results presented in this paper will generalize to more realistic noise conditions and to feature representations other than the MFCCs used in these experiments.

6. REFERENCES

[1] S. Kullback, *Information Theory and Statistics*, Wiley, New York, 1959.

[2] M. Basseville, "Distance measures for signal processing and pattern recognition," *Signal Processing*, vol. 18, pp. 349–369, 1989.

[3] E. A. den Os, T. I. Boogaart, L. Boves, and E. Klabbers, "The Dutch Polyphone corpus," in *Proceedings of Eurospeech '95*, Madrid, 1995, pp. 825–828.

[4] S. Young, J. Jansen, J. Odell, D. Ollason, and P. Woodland, *The HTK Book (for HTK Version 2.1)*, Cambridge University, Cambridge, UK, 1995.

[5] B. Noe, J. Sienel, D. Jouvet, L. Mauuary, L. Boves, J. de Veth, and F. de Wet, "Noise reduction for noise robust feature extraction for distributed speech recognition," in *Proceedings of Eurospeech '01*, Aalborg, 2001, pp. 433–436.