IMPROVING AUTOMATIC SPEECH RECOGNITION ROBUSTNESS FOR THE ROMANIAN LANGUAGE

Andi Buzo, Horia Cucu, Corneliu Burileanu

Faculty of Electronics, Telecommunications and Information Technology,
University “Politehnica” of Bucharest
“Iuliu Maniu” 1-3, Sector 6, Bucharest, Romania
phone: +40 744 700021, cburileanu@messnet.pub.ro

ABSTRACT

In this paper we propose an alternative way of improving speech recognition accuracy by analyzing the relevance of voice feature dimensions. A new measure is defined in order to quantify the feature distribution overlapping. Based on this measure, weights for voice feature dimensions are calculated and then applied to the hypotheses resulted from an N-best recognition process. Experiments are made with an Automatic Speech Recognition (ASR) system for the Romanian language. A relative improvement of 22% is obtained in terms of Word Error Rate (WER).

1. INTRODUCTION

We propose a two-step speech recognition process with improved recognition accuracy. The first step is a classical Hidden Markov Model (HMM) recognition process with *N*-best hypotheses. In the second step, the *N*-best hypotheses are compared frame-by-frame and the voice features are weighted based on a measure of uncertainty, defined as the overlapping of the trained feature distributions between the acoustic models. Feature dimensions with smaller uncertainties are given greater weights.

Several previous research efforts have dealt with the uncertainty that derives from feature enhancement/noise removal, by using the missing data algorithms and uncertainty decoding, with the final goal of weighting the feature dimensions according to their reliability [1], [2]. Instead, here, the speech features are weighted based on the acoustic *model uncertainty*, i.e., the inherent overlapping of different state output probability density functions (pdfs). Hence, this can be used either as an alternative technique or incorporated in the data missing and uncertainty decoding algorithms, where acoustic *model uncertainty* gives an indication about the feature relevance. The uncertainty of the acoustic models is also used in [3] and [4] and Support Vector Machines are used to define hyperplanes (as model boundaries) that separate model classes. In a similar study [5], voice feature dimensions contribution to the final score is weighted in every speech frame based on the entropy that the feature dimension presents. This involves the calculation of the entropy at every frame, while in our approach the weights can be pre-computed since they are calculated using the trained models and they are independent from the speech observation. *N*-best speech recognition hypotheses are often used when there is supplementary information that helps to choose the best option, usually by applying language models as in [6] and [7], but also by using confusion rates as in [8]. In our approach the recognition accuracy is improved by applying weights to the voice feature dimensions according to their relevance.

2. THE PROPOSED APPROACH

The first step consists in a classical *N*-best ASR process based on HMMs. The recognition accuracy in terms of Word Error Rate (WER) for different values of *N* is shown in Figure 1. The characteristics of the ASR system and of the database which give these results are described in Section 3. In Figure 1, WER is calculated by considering as an error the case in which the correct word is not found in any of the *N* hypotheses. The performance of the system without any other improvement is given by the value of WER for *N* = 1 (18.5%). This means that we can potentially decrease WER by finding a way of making better decisions among the *N*-best hypotheses.

In traditional ASR systems where a Gaussian Mixture Model (GMM) is used to model the HMM state output, the log-likelihood for each state *j* is given by equation (1):

$$\log(b_j(O_t)) = \log(\sum_{g=1}^{G} C_{jg} N(O_t, \mu_{jg}, \Sigma_{jg})) \, . \qquad (1)$$

where O_t is the observation vector at time t, $N(O_t,\mu_{jg},\Sigma_{jg})$ is the multivariate Gaussian, C_{jg} is the weight of the mixture-component and G is the number of mixture components.

In this case all the feature dimensions have the same weight in the final score. However, not all the feature dimensions discriminate models in the same way. Figure 2 shows an example of how feature distributions for two different models overlap: f(x) and g(x) are the Gaussian mixtures for model *Mi* and *Mj* respectively, pdf values are represented on the ordinate axis. The greyed area represents *P(Dj|Mi),* the probability of erroneously deciding that model *j* is detected instead of model *i.* Similarly, *P(Di|Mj)* is the probability of erroneously deciding that model *i* is detected instead of model *j.* We define this overlapping (both *P(Dj|Mi)* and *P(Di|Mj)* areas) as *model uncertainty* and the notation used for it is *u.* The greater the *model uncertainty*, the greater the

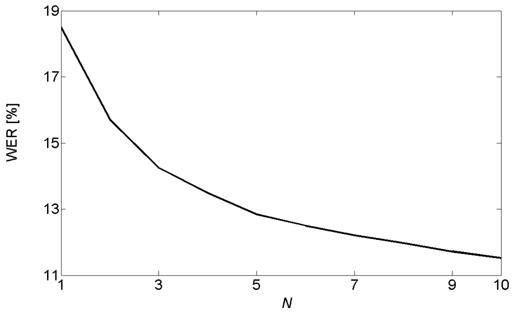

Figure 1 - The variation of WER with the number of hypotheses *N*.

probability to make erroneous decisions and the lower the capacity of the dimension to discriminate the models, thus, the feature dimensions with lower *model uncertainty* are more reliable.

Model uncertainty (*u*) can be calculated as the numeric integration of function h(x) defined as:

$$h(x) = \begin{cases} f(x), f(x) < g(x) \\ g(x), g(x) < f(x) \end{cases}. \tag{2}$$

whereas the numeric integration is calculated with the trapezoid rule as shown in equation (3):

$$u = \int_{-\infty}^{\infty} h(x)dx \cong \int_{a}^{b} h(x)dx = \\ = \sum_{k}^{n-1} [\frac{h(a+kd) + h(a+(k+1)d)}{2}]. \tag{3}$$

where *n* = (*b*-*a*)/*d*.

In our implementation we used: *a* = -100, *b* = 100, *d* = 0.01, which yield a calculation error lower than 10^{-6}. This operation is made offline, hence, *a, b* and *d* can be chosen in order to minimise the calculation error. Eventually, *model uncertainty* is calculated for every combination of two different states.

In the hypothesis that the decision is made only between two GMMs, the feature dimensions can be weighted according to their *model uncertainty* (equation (4)). Hence, the error deriving from the acoustic uncertainty is minimised.

By applying the weights, the formula for the calculation of the log-likelihood per frame becomes:

$$\log N(O,\mu,\Sigma) = -\frac{1}{2} n \ln 2\pi - \\ -\frac{1}{2}\sum_{i}^{n} w_i [\frac{(o_i - \mu_i)^2}{\sigma_i^2} + \sigma_i^2]. \tag{4}$$

where w_i are the weights and *Σ* is the diagonal covariance matrix.

Because feature dimensions with smaller *model uncertainty* must have greater weights, a monotonically decreasing function is needed in order to calculate weights from *model uncertainty*. In this paper, two such functions are used for this purpose, presented in equations (5) and (6):

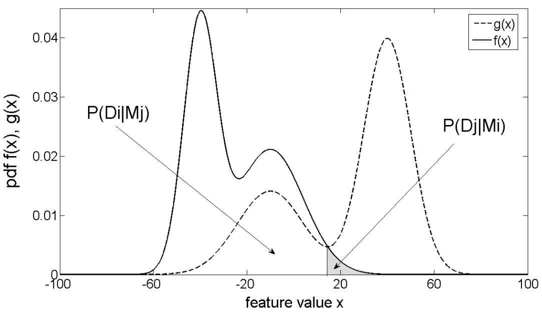

Figure 2 - Definition of *model uncertainty* as the overlapping between two distributions of different states.

$$w_{it} = \frac{(u_{it})^{-1}}{\sum_{k=1}^{n} (u_{kt})^{-1}}. \tag{5}$$

where u_{it} is the *model uncertainty* for feature dimension *i* at time frame *t*, whereas the denominator is used for normalization.

$$w_{it} = \frac{\exp(-a \cdot u_{it})}{\sum_{k=1}^{n} \exp(-a \cdot u_{kt})}. \tag{6}$$

where *a* is a tuning parameter, and the denominator is used, here as well, for normalization.

In equation (5), the weight is inversely proportional to the *model uncertainty*, whereas in equation (6), the weight varies exponentially with the *model uncertainty*. Experiments have shown that when weights are calculated with equation (5), there is big difference between them and very often the decision is made only by a few feature dimensions. This effect is smoothed by using the equation (6). The tuning parameter *a* helps controlling the range for the weights.

When deciding among a great number of GMMs, the calculation of weights (as in equations (5) and (6)) is not straightforward, because one GMM can have different *model uncertainties* with different GMMs. According to the way *model uncertainty* is defined, it can be applied only when deciding between two hypotheses. For this reason, the weights are applied only to the *N* hypotheses resulted from the *N*-best ASR process. Hypotheses are grouped in pairs, and from each pair a winner is elected for the next round of comparisons until a final winner is elected. Hence, it would be best for *N* to be equal to a power of 2, but the method can work with any value of *N*. The first hypothesis is paired with the last one, the second is paired with the one before the last, and so on.

From the *N*-best ASR process, the succession of states is provided, therefore, the GMMs which are going to be compared, are also known. The hypotheses being compared within a pair are aligned, and the pre-calculated weights are applied for each frame. In Section 3, this method will be referred to as **Method 1**.

As an alternative to the log-probability, another score is used in this paper. The scores (log-probability) per frame are calculated in the same way, while the final score for a hypothesis counts the number of frames in which it has a better score than the other hypothesis from the same pair. This method gives good results with degraded speech signals where the scores of some degraded frames can affect the overall score. In Section 3, this method will be referred to as **Method 2** and it is shown that it gives good results even with clean speech signals.

The added complexity is low because the model uncertainties are calculated offline, and during the decoding process, the terms of the summation in the log-likelihood formula (equation (4)) are only weighted. A table with the model uncertainties must be stored in the memory. Its dimensions are KxK, where K=(No. phones)x(No. HMM states).

3. EXPERIMENTAL RESULTS

Experiments are made with an ASR system using a database in the Romanian language. 5 speakers have pronounced the same 10000 distinct words for a total of 50000 words. 45000 words are used for training and 5000 for testing. An important property of the database is that it contains all the syllables of the Romanian language. HMMs are used for modelling the phonemes. Each model has 12 states and state i can only transit to states i, i+1 or i+2. The output of the states is a linear combination of 3 Gaussian functions. Mel-Frequency Cepstrum Coefficients (MFCC), including energy, are used as voice features (13 coefficients), along with their first-order derivatives, for a total of 26 dimensions.

The HTK ToolKit [9] is used for both training and recognition. Embedded training with the Baum-Welch algorithm is used for obtaining the trained models, whereas the Token Passing [10] algorithm is used for decoding in the first step. The N-best hypotheses obtained in the first step are then grouped in pairs (as shown in Section 2) and compared by the log-probability calculated with **Method 1**. The results are presented in Table 1. More details about the database and the set-up used can be found in [11].

Experiments are made for different values of N. The *Maximum Score* column shows the potential improvement that can be made by using the N-Best technique. If the correct word is found in one of the N hypotheses, then the recognitions is considered correct. The rest of the columns show the WER for **Method 1**, where weights are calculated with equations (5) and (6) respectively and three different values for the tuning parameter a are used. The weights that are calculated as inversely proportional to the *model uncertainties* do not bring any improvement in terms of WER. However, not all the words recognized correctly in this case are the same with the words calculated correctly in the first step. For example, for N = 4, 2% of the words are correctly recognized in the first step and erroneously recognized in the second step. These results can be explained by the way weights are calculated, i.e. as inversely proportional to *model uncertainties*. This method can lead to high differences among weights and in some cases the decision is practically made only by a few feature dimensions. If the weights are calculated exponentially, the problem can be solved and, in this manner, better results are obtained. The tuning parameter a determines the sensitivity of weights to the *model uncertainties*. For greater values of a, the weights vary in a larger range, whereas for smaller values of a, the weights tend to be equal. Hence, an optimal value for a is expected. In this paper, the best values are obtained for a = 0.5.

Table 1 – The results obtained with **Method 1**

N	*Maximum score*	WER [%] with inversely proportional function	WER [%] with exponential function, a = 1	WER [%] with exponential function, a = 0.5	WER [%] with exponential function, a = 0.1
1	18.50	18.50	18.50	18.50	18.50
2	15.71	18.63	17.46	16.52	17.61
3	14.25	18.53	16.97	15.34	17.15
4	13.49	18.57	16.56	15.03	16.94
5	12.84	18.52	16.38	14.85	16.81
6	12.50	18.51	16.32	14.65	16.72
7	12.21	18.49	16.25	14.61	16.61
8	11.97	18.48	16.14	14.56	16.56
9	11.72	18.46	16.10	14.54	16.52
10	11.52	18.47	16.08	14.48	16.48
100	8.44	18.51	15.94	14.42	16.46

Table 2 – The results obtained with **Method 2**

N	*Maximum score*	WER [%] with inversely proportional function	WER [%] with exponential function, a = 1	WER [%] with exponential function, a = 0.5	WER [%] with exponential function, a = 0.1
1	18.50	18.50	18.50	18.50	18.50
2	15.71	18.63	17.21	16.26	17.45
3	14.25	18.53	16.62	15.05	16.91
4	13.49	18.53	16.26	14.71	16.71
5	12.84	18.52	16.00	14.51	16.48
6	12.50	18.53	15.91	14.29	16.41
7	12.21	18.47	15.84	14.23	16.32
8	11.97	18.47	15.78	14.18	16.24
9	11.72	18.46	15.73	14.12	16.22
10	11.52	18.47	15.72	14.09	16.19
100	8.44	18.50	15.68	14.04	16.16

WER decreases when N increases, but for greater values of N the WER will be saturated. Greater values of N also imply more processing power and are time-consuming. So the appropriate value for N is chosen based on the system resources. A relative improvement of 22% is obtained in terms of reducing WER.

The results obtained with **Method 2** are presented in Table 2. They are similar to, yet slightly better than the results obtained with **Method 1**. This can be explained by taking into consideration that for some phonemes of the correct word the scores obtained in the respective frames are very low which is affecting the overall score, and may be in favour of another similar word. In the case of exponential function with a = 0.5, WER is reduced with a relative value of 2.7%.

Compared to a similar study [5], where an average error reduction of 15.4% is obtained, our results seem promising. However, it must be mentioned that in [5], it is used a different database i.e. AURORA 2 testing environment [12] based on a corpus of English connected digit strings. The words of our database are chosen so that all the syllables of the Romanian language are covered. Each word in the database has only one occurrence. Consequently, the words of the testing database are totally different from the words of the training database. Moreover, there are some phonemes that have very few occurrences (less than 100).

4. CONCLUSIONS

In this paper, an improvement of ASR accuracy, for the Romanian language, in terms of WER is achieved by weighting the voice feature dimensions according to their relevance. *Model uncertainty* is defined as a measure of the distributions overlapping for two feature dimensions. Two functions are used to deduce weights from the *model uncertainties*. First, the weights are calculated according to a function that is inversely proportional to the *model uncertainties.* Secondly, an exponential function is used. Weights are applied during the comparison of the hypotheses resulted from a classical *N*-best ASR process. Two methods are used for the calculation of the log-likelihood, the overall score and the per frame score.

The experimental results show that the application of weights yields a relative WER reduction of 22%. A comparison between calculation modes for weights has been made and the exponential function resulted to be better than the inversely proportional function which did not bring any improvement to the ASR accuracy. Also, the calculation of scores per frame gives better results than the calculation of an overall score, with a relative value of 2.7%.

5. ACKNOWLEDGEMENTS

The research reported in this paper was funded by the Romanian Government, under the National Research Authority CNCSIS grant “IDEI” no. 114/2007, code ID_930 and by the Sectoral Operational Program Human Resources Development 2007-2013 of the Romanian Ministry of Labor, Family and Social Protection through the Financial Agreement POSDRU/6/1.5/S/16.

REFERENCES

[1] J. Droppo, A. Acero, L. Deng, “Uncertainty decoding with splice for noise robust speech recognition”, Proc. ICASSP, pp. 57-60, 2002.

[2] H.Liao, M.J.F. Gales, 2005. “Joint uncertainty decoding for noise robust speech recognition”, Proc. INTERSPEECH, pp.3129-3132, 2005.

[3] J. Hamaker, J. Picone, A. Ganapathiraju, “A Sparse Modeling Approach to Speech Recognition Based on Relevance Vector Machines”, Proc. ICSLP, vol. 2, pp. 1001-1004, 2002.

[4] J. Bi, T. Zhang, “Support vector classification with input data uncertainty”, Proc. NIPS, pp. 161-168, 2004.

[5] Y. Chen, C. Wan, L. Lee, “Entropy-based feature parameter weighting for robust speech recognition”, in Proc. ICASSP, pp. 41-44, 2006.

[6] R. Zhang, G. Kikui, H. Yamamoto, F. Soong, T. Watanabe, E.Sumita, W.Lo, “Improved spoken language translation using n-best speech recognition hypotheses”, Proc. INTERSPEECH-2004, pp. 1629-1632, 2004.

[7] B.H. Tran, F. Seide, V. Steinbiss, R. Schwartz, Y.L. Chow, “A word graph based N-best search in continuous speech recognition”, Proc. ICSLP, pp. 2127-2130, 1996.

[8] K. Kim, H. Kim, M. Hahn, “Utterance Verification Using Search Confusion Rate and Its N-Best Approach”, ETRI Journal, vol. 27, no. 4, pp. 461-464, 2005.

[9] G. Everman, et al., “The HTK Book”, Version 3.0, Cambridge University, Engineering Department, 2005.

[10] S.J. Young, N.H. Russell, J.H.S. Thornton, “Token passing: a simple conceptual model for connected speech recognition systems”, Technical Report, Cambridge University: Engineering Department, 1989.

[11] C. S. Petrea, A. Buzo, H. Cucu, M. Paşca, C. Burileanu, “Speech Recognition Experimental Results for Romanian Language”, Proceedings of ECIT2010 – The 6th European Conference on Intelligent Systems and Technologies, Iasi, Romania, 2010, CD, 13p, Invited Plenary Session I.

[12] H.G. Hirsch, D. Pearce, “The AURORA Experimental Framework for the Performance Evaluation of Speech Recognition Systems under Noisy Conditions”, Proc. ISCA ITRW ASR2000, 2000.