IMPROVED SPEAKER VERIFICATION SYSTEM WITH LIMITED TRAINING DATA ON TELEPHONE QUALITY SPEECH

S. Hussain†‡, F. R. McInnes†and M. A. Jack†
e-mail: hussain@ccir.ed.ac.uk
†Centre for Communication Interface Research, University of Edinburgh, UK
‡Faculty of Electrical Engineering, Universiti Teknologi Malaysia, Malaysia

ABSTRACT

A hybrid neural network is proposed for speaker verification (SV). The basic idea in this system is the usage of vector quantization preprocessing as the feature extractor. The experiments were carried out using a neural network model(NNM) with frame labelling performed from a client codebook known as NNM-C. Improved performance for NNM-C with more inputs and proper alignment of the speech signals supports the hypothesis that a more detailed representation of the speech patterns proved helpful for the system. The flexibility of this system allows an equal error rate (EER) of 11.2% on a single isolated digit and 0.7% on a sequence of 12 isolated digits. This paper also compares neural network speaker verification system with the more conventional method like Hidden Markov models.

1 INTRODUCTION

Various approaches have been developed for the whole word based SV task. In these cases word variations may occur especially in repeating the right intonation. One form of variability that can affect the word based recognition is the non linear compression and expansion of the speech signal from one word to the other. For example, an efficient procedure relies on the application of dynamic programming. Dynamic Programming Neural Networks(DNN) have been proposed for speech recognition tasks using dynamic programming(DP) and multi layer perceptron. The input units are arranged in a block structure frame along the time axis. The input pattern is optimally time aligned by DP so that the output unit gives maximum output. A speaker independent isolated Japanese digit recognition experiment was carried out with 107 speakers resulting in 99.3% recognition accuracy. This DNN makes use of the valuable feature of time normalization of DP and classification power of NNs[1]. Another procedure to normalize the length of the word is the use of trace segmentation. This procedure was used by Zhu and Fellbaum[2] to compress nonlinearity of the speech signals into a fixed length of 24 triples. The outputs of the MLP give the classification results. The system was capable of achieving 95% accuracy from 22 German words used in the experiments. However, one of the simplest approach to overcome the variability of the word is the linear time normalization(LTN). LTN is made to correspond as closely as possible to a straight line joining the initial and final points. The approach of using LTN is implemented in this paper for the verification system. The problem with LTN is that phonetic events (especially short time events) such as the plosive can be discarded during the process or insertion of feature vector may alter its relative duration. This is more important than the steady state information in the vowel. Care must be taken in selecting the proper LTN values which preserves the nonlinearity of the speech signals. It is of interest to study the effects of different values of LTN (with respect to information being discarded or inserted) to the speaker verification performance.

2 SPEAKER VERIFICATION METHOD

Oglesby and Mason [3] proposed a text dependent speaker verification(SV) using one MLP per speaker. These approaches used raw features to feed into the neural network. The larger and more complex the input space the more training samples are needed for training before the network can learn to generalize. There is also the possibility that large number of hidden nodes are required to solve the problem. If this is the case then training may be difficult as not only will the MLP takes a long time to learn but it will also increase the chance to get trapped in a local minimum which may not yield a good solution to the problem.

In the method described in the present paper, vector quantization is used in a preprocessing stage to reduce the number of input features. A self organization network is combined with the LBG technique to design the vector quantizer. Once the codebook is generated, the preprocessing stage uses the vector quantizer to select the index. The indices of the winner nodes are fed to a neural network classifier in which the system can be trained and evaluated. The use of a preprocessing stage allows a smaller network configuration. This can eliminate the difficulties in the

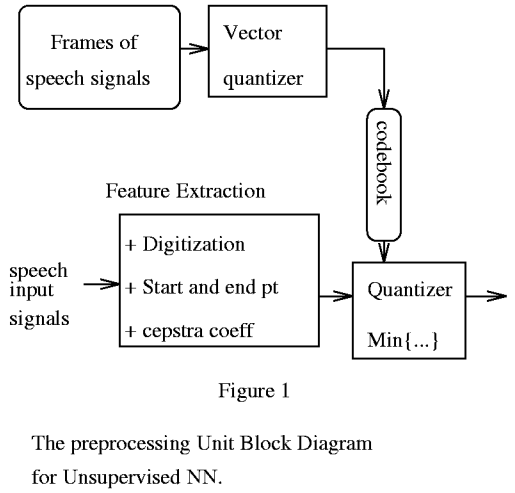

Figure 1

The preprocessing Unit Block Diagram for Unsupervised NN.

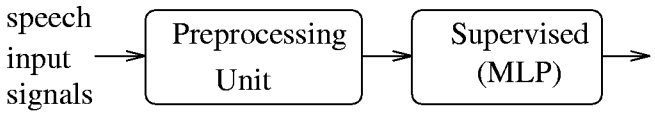

Figure 2

A neural networks based speaker verification system

training phase and facilitates training on limited data. The verification system is shown in Figures 1 and 2. The initial stage used the commonly used feature set, cepstral coefficients for the speaker modelling stage as the speech signals. For each frame of the input speech, the output of the preprocessor would contain the index j of the codevector with the minimal distortion and the corresponding distortion value d. The input pattern is linear time normalized(LTN) either by linear compression or expansion so that the total number of frames becomes a constant regardless of the word duration. Through the preprocessing stage, the highly redundant speech data are reduced so that only the useful information regarding codevector and the distance measure is retained in the feature vector to feed the MLP. For example, if the number of frames after LTN=40, two coefficients per frame will fit the 80 input units. By using the client codebook each hidden unit is fed with 80 input units resulting in an architecture of 80-N-1 where N is the number of hidden units. The classifier system is based on a three layer perceptron trained using the back-propagation algorithm. The training scheme used a separate net for each digit for each speaker. Separate nets were trained for each of the 12 digits for each of the 11 speakers.

In the previous work[4] a comparison was made between two alternative approaches for speaker verification(SV) using neural network. The first experiment used a neural network model(NNM) with frame labelling performed from a client codebook known as NNM-C. The second set of experiments used the NNM with frame labelling from the client and the impostors codebook known as NNM-CI. The NNM-C performs better than the NNM-CI in all the digits. It is reasonable to conclude from the results that NNM-C model should be used in preference to NNM-CI model when training with limited data. Using the NNM-C also means the amount of input to be fed to MLP is reduced by 50%.

3 SPEECH DATA

The data base consists of the isolated digits from a large number of speakers. Twelve isolated digits(digits 'one' to 'nine' plus 'zero','nought' and 'oh') were used in the experiments. A group of 11 speakers are modelled by the system and an independent set of 83 impostor speakers is used for testing. The data are all end-point detected to remove excess silence and minimize storage requirements. The framesize was 20ms with 15ms overlap. The training templates consisted of 5 tokens from the client speaker and 19 from the impostors (different from the impostors used in testing). The templates from the target group and the impostor group were alternated in the training set. The implemented verification system used another set of data (not used during training) for further evaluation of its performance. It was tested on 20 true speaker tokens and 83 impostor tokens for each digit for each speaker. In the evaluation of the verification system the use of equal error rate (EER) thresholds means that all thresholds are determined a posteriori. This approach sets the proportion of false acceptance equal to the proportion of false rejection resulting in the said EER.

4 EXPERIMENTAL RESULTS

If the standard net configuration is fed with 40 feature vectors of 12 cepstrum coefficients, then this standard architecture will have 480 input units. For the 40LTN architecture the number of links from the input layer to the hidden layer is reduced by a factor of 6 in comparison with the standard architecture mentioned above. In this section the performance of the NNM-C model with other LTN values is evaluated. The lengths of inputs after LTN are 30LTN, 50LTN and 60LTN. Considering the very coarse data where the number of links has been reduced by a factor of 4, 5, 6, and 8 there is evidence from the results that the

approaches used are well suited for the speaker verification task trained on limited training data.

4.1 Single Digit Performance With Different LTN Values

Digit	EER			
	LTN30	LTN40	LTN50	LTN60
1	13.6	13.5	11.9	9.1
2	16.8	14.3	15.0	12.1
3	15.0	13.1	14.3	11.0
4	17.1	15.0	15.6	13.5
5	11.2	10.0	10.6	8.7
6	18.2	15.3	16.5	13.5
7	17.9	14.4	16.4	14.9
8	17.8	16.2	17.6	12.2
9	10.9	6.9	10.5	8.2
zero	10.6	8.9	7.8	6.6
nought	18.3	15.6	17.1	14.5
oh	13.5	12.9	13.1	10.7
mean	15.0	13.0	13.9	11.2

Table 1.0: Performance over single digit results of the four LTN inputs with speaker specific threshold.

In this section the performance of NNM-C model with different LTN values is evaluated on a single isolated digit test utterance. The EER was evaluated for each of the 12 digits. The performance of each of the digits used for the different LTN value is listed in Table 1.0. From the table shown there is considerable variation in performance across the digits. LTN30 has an average EER of 15.0% with a range from 11% to 18%. LTN40 has an average EER of 13.0% with a range from 7% to 16%. LTN50 has an average EER of 13.9% with a range from 8% to 18%. As for LTN60, it has an average EER of 11.2% with a range from 7% to 15%. The best digit across the range of the utterances is zero, while digits like 1, 5 and 9 also show good performance results. The worst performances are for digits 4, 6, 7, 8 and 'nought'. According to Yu, Mason and Oglesby, the good performance of the digit zero could be attributed to it being the longest utterance thus containing more information as well as the voiced fricative of the first phoneme being a particularly useful phoneme in speaker recognition[5]. While the digits have different performance on their own, each digit emphasises different aspects of the time varying speech signals and the rankings of the digits may vary from client to client. The fact that specific digits can significantly improve performance indicates that a password system consisting of these digits could be found to suit each client speaker.

4.2 Digit Sequence Results With Different LTN Values

The EER performance for each of the LTN inputs over various digit sequence lengths is shown in Figure 2.0. As more and more digits are being added, the speaker discriminative information it contains becomes more apparent as shown in figure. In most LTN values chosen it appears to stabilize at around digit 10. However, this is not the case for LTN30 which shows signs of instability. This is probably due to too much compression of the speech signals and some essential information may be lost which results in poor generalization of the NNM-C. An increasing number of input units does affect the verification performance but does not necessarily bring about an improvement in performance. The results suggested proper values of time normalization of the speech signals are also needed besides the features used for better performance of the speaker verification system. Ideally, the knowledge stored in the hidden layer of the NNM-C model is abstracted from the information contained in the LTN input speech patterns. This abstracted knowledge provides the basis for the model to classify the pattern into an acceptable category or a reject at the output unit.

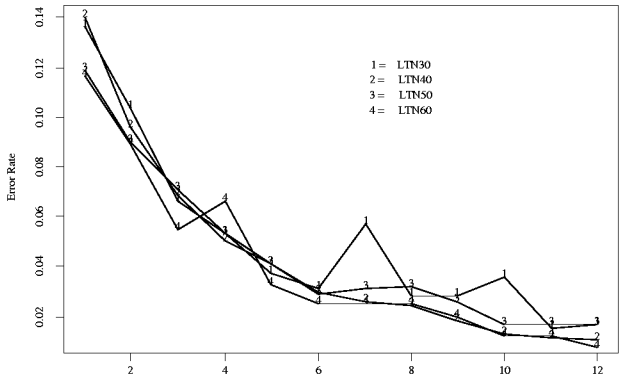

Figure 2.0: Performance over digit sequence results of the four LTN inputs.

LTN	EER	%error relative to best LTN
30	1.75	60
40	1.04	32.6
50	1.7	58.8
60	0.7	-

Table 2.0: Single LTN set results. Speaker specific EERs are given for the 12 digit sequence.

The relative performance for different LTN values for the 12 digit sequence is given in Table 2.0. It can be seen that in all cases there is different error rate.

Percentage error is the additional error obtained by using the LTN value when compared with the best LTN value. It can be seen from the table that for the 12 digit sequence LTN60 has speaker specific EER of 0.7% compared to 1.75% for LTN30. This gives the indication that increasing number of inputs supports the verification performance for better results. On the other hand an EER of 1.04% is achieved with LTN40 compared to 1.7% for LTN50. The best performance result achieved for NNM-C is LTN60.

5 COMPARISON WITH THE ESTABLISHED TECHNIQUE HMM

When comparing the work carried out in this paper with other published results it is important to consider that different systems are also trained on a limited number of training tokens and the performances of the systems are evaluated on the same data base. Comparison of different systems which use different amount of training data and different data base are not very meaningful. In view of this, a comparison is made with the established technique of HMM [6] applied to the same data base. One difference in Forsyth's system from the NNM is that a standard codebook is used for all speakers and for all digits instead of different codebooks for all speakers and digits. Another difference is that the HMM system was trained to model only the client data rather than to discriminate between client and impostor data. For DHMM(Discrete Hidden Markov Models), models with 3 and with 6 states were constructed for each digit. SCHMM(Semi Continuous Hidden Markov Model) with 6 states was constructed for each digit. Both of these models are trained with 5 and 10 training tokens and tested with 10 true client tokens and 95 or 100 impostor tokens for each digit for each speaker. Neural networks SV on the other hand was trained with 5 client tokens and 19 impostor tokens and tested on 20 true client tokens and 83 impostor tokens. EER for individual digits for DHMM ranged from 12%-28% for the 5 token models and 8%-17% for the 10 token models. Average EER of 14%(DHMM) and 12%(SCHMM) were achieved for single isolated digits and 4%(DHMM) and 2%(SCHMM) for a sequence of 12 isolated digits for the 10 token models. Neural network SV experiments, with 60LTN, produce an EER of 0.7% compared to 4% for DHMM trained with 10 tokens. This favours the neural network approach considering the differences in the amount of training data. Note also the performance of Forsyth's system continues to drop to 2% with the SCHMM models. Neural networks using 5 training tokens produced an average EER of 11.2% compared to 18.5%(DHMM) and 14.3%(SCHMM) for single isolated digits.

6 DISCUSSION

The approach of having a fixed input to the NNM-C is one of the simplest methods of time aligning in a linear fashion of the speech signals. One advantage of this approach is that it does give proper alignment of the beginning and the end of the patterns. Improved performance for NNM-C with more inputs and proper alignment of the speech signals supports the hypothesis that a more detailed representation of the speech patterns proved helpful for the system. This paper established the relative performance of the different LTN values used in the experiments. It also suggests the possibility of selecting the best LTN values in order to improve the robustness of the NNM-C model.

Finally, it can be concluded that reducing the input vectors to a bearable size still allows the classification power of neural networks to discriminate between the client and the impostor speakers. This demonstrates the usefulness of the preprocessing stage in the design of ASV system. The NNM-C model shows a significant improvement over conventional HMM speaker verification system.

REFERENCES

[1] Sakoe, H., Isotani, R., Yoshida, K., Iso, K., Watanabe, T. 1989 (May). "Speaker-Independent Word Recognition Using Dynamic Programming Neural Network". Pages 29-32 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing.

[2] Zhu, M., Fellbaum, K. 1990 (May). "A Connectionist Model For Speaker-Independent Isolated Word Recognition". Pages 529-532 of: Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing, vol.2

[3] Oglesby, J., and Mason, S. : "Optimization of Neural Models For Speaker Identification", Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing, vol.1, April 1984, pp. 261-264.

[4] Hussain, S., McInnes, F.R., Jack M.A. 1996 (December) "Comparison of Neural Network Techniques For Speaker Verification". Pages 245-250 of: Sixth Australian Conference Speech Science Technology.

[5] Yu, K., Mason, J., Oglesby, J. 1995 (September). "Speaker Recognition Models". Pages 629-632 of: Proceedings of the European Conference on Speech Technology, vol.1.

[6] Forsyth, M., Sutherland, A., Elliott J., and Jack, M. : 'HMM speaker Verification with sparse training data on telephone quality speech', Speech Communication, 1993, Vol.13, pp. 411-416.