Proceedings of The Institute of Acoustics

DYNAMIC SPEAKER ADAPTATION IN SPEAKER-INDEPENDENT WORD RECOGNITION

A. J. Hewett, G. Holmes and S. J. Young.

Cambridge University Engineering Department.

INTRODUCTION

In speaker-independent speech recognition we are faced with the problem of trying to recognise speech from an unfamiliar speaker. One approach to this problem using Template Matching is to collect examples of a vocabulary from many speakers and rely on statistical classification techniques, such as clustering, to identify prototypical patterns which can be used in recognition [1]. Another technique uses this data to train the parameters of a Hidden Markov Model for each vocabulary item [2]. Within a speech recognition system conducting a dialogue, a large amount of information is available relating to the current speaker's characteristics. In their standard forms, the application of both Template Matching and Hidden Markov Model methods are ignoring the consistency of speaker. This is equivalent to assuming that each utterance in the dialogue is spoken by a different speaker.

Speaker adaptation has often been suggested as a technique for improving performance when recognising speech during a dialogue with a single speaker. Green et al [3] have investigated the behaviour of Template Matching systems which retrain their reference templates dynamically during recognition by averaging with the recognised speech. They found that quite substantial improvements in recognition accuracy could be obtained provided that the updating process was stable, i.e. that incorrect recognitions did not corrupt all the reference templates for the incorrectly recognised word. Damper & MacDonald [4] also investigated template adaptation through retraining, reaching the same conclusions regarding the importance of stability within adaptive systems.

The type of recogniser we are considering uses multiple reference templates to represent each vocabulary item which are then compared with the unknown speech using a Dynamic Programming (DP) algorithm [5]. Training involves selecting the reference templates from training data spoken by many speakers using a cluster analysis algorithm [6]. Such algorithms attempt to partition the data into a number of clusters such that members of each cluster are similar in some respect. Each cluster can be thought of as representing speaker subpopulations within the data, although many clusters may not correspond to any obviously identifiable categories. Thus, speaker variation is modelled by prototypical patterns covering the entire speaker space.

The work we report here is concerned with speaker-independent word recognition and is quite different to the adaptation through retraining approach of Green et al which was essentially a speaker-dependent word recogniser which integrated training with recognition. A typical example of a speaker-independent recognition application is a database inquiry system operating over the telephone network. Any user dialing in will be unknown to the system and for speaker adaptation to be effective, it must operate during the lifetime of a single dialogue. In such cases there is little opportunity for retraining during recognition. Instead, we investigate the possibilities for adapting the recogniser's choice of reference templates to the current speaker.

Our basic premise is that only a subset of the entire reference template inventory is actually needed when recognising speech from a single speaker. Not only would this lead to a reduction in computation and memory requirements but it would seem likely that using only the reference templates applicable to a particular speaker could improve recognition accuracy through a reduction in cross-talker confusability. As noted before, to be practical, the approach must be capable of operating within a quite short time scale.

SOURCES OF FEEDBACK

A major factor mentioned above, which is critical to the performance of an adaptive recognition system, and indeed any adaptive system, is stability. There are, for our purposes, two major sources of error feedback by which adaptation can be effected and stability ensured, external and internal feedback. External feedback derives from any source outside the low level recognition component. Typical sources would be the higher level syntactic, semantic and pragmatic components in a hierarchical speech understanding system. However, we only consider here the simpler internal feedback but note that external feedback could be an important control source. For the template matching word recogniser we use here, our only source of internal feedback is the 'goodness of fit' scores generated by the DP algorithm. These scores do not however, provide a convenient framework on which to base adaptive decisions. Instead, we transform these scores into likelihood estimates. Unfortunately, words will occasionally be recognised incorrectly but with high likelihood due to the nature of the pattern matching process. This can, in turn, cause mis-adaptation and stability problems. We must therefore provide some means to recover from incorrect decisions. Without any external error feedback this is a very difficult situation to detect, especially in the adaptation method used here, since our whole purpose of reducing the number of reference templates also reduces the amount of internal feedback available to us. We return to this point in the following sections.

ADAPTIVE DECISION CLASSES

At any time during recognition a member of the reference template inventory may be either active (useful for the current speaker) or inactive (not considered useful to the current speaker). After each recognition cycle we have the opportunity to deactivate or reactivate members of the inventory depending on current performance. The speed with which adaptation can take place is greatly influenced by the knowledge of relationships between subsets of the reference template inventory. The more knowledge that can be invoked, the quicker we can adapt. The application of these knowledge sources can be organised into Adaptive Decision Classes (ADC's). In general, each ADC can operate on subsets S_1 S_2 .. S_n of the reference template inventory I. Furthermore, each ADC has associated with it two decision rules $R_-(S_i)$ and $R_+(S_i)$ which specify how adaptation can take place. The negative decision rule $R_-(S_i)$ defines when the

members of a reference template subset S_i can be made inactive and conversely, the positive decision rule $R_+(S_i)$ defines when S_i can be reactivated. The decision rules must also operate according to the following constraints

1. $R_-(S_i)$ cannot be accepted if it would result in leaving no active reference templates for a vocabulary item.

2. $R_+(S_i)$ can only be accepted if S_i is the most recently deactivated subset still inactive.

3. At any recognition cycle only one of $R_-(S_i)$ and $R_+(S_i)$ can operate on only one of the possible S_i

Constraint 1 ensures that every vocabulary item has at least one reference template representing it. Constraint 2 ensures that the most recent decisions are undone if enough evidence suggests that they were wrong, while constraint 3 acts as an aid to stability by limiting the effect of decisions during any single recognition cycle.

While these decision rules are, at the moment, largely heuristic, they should ideally display some general properties. Firstly, the amount of certainty required before accepting a decision should be proportional to the power of the decision. For example, if an ADC contains only two subsets, S_1 and S_2 which allow a binary decision on I, then the negative decision rule $R_-(S_2)$ must display a high degree of certainty before accepting S_1 and deactivating S_2. A second property follows from the first in that the certainty, and hence the amount of information, required to accept $R_+(S_i)$ should be equivalent to the information needed before $R_-(S_i)$ can be accepted. However, as mentioned before, deactivation of reference templates naturally leaves us with less information on which to base decisions, and so the equivalence between rules can only be approximate.

LIKELIHOOD ESTIMATION

The direct use of distance scores to provide internally generated error feedback suffers from a number of problems. Firstly, it is very difficult to use such distances as a measure of reliability (i.e. is a distance of 10765 good?). Secondly, some words naturally generate relatively higher distances due to their varying phonetic makeup. Clotworthy & Smith [7] have recently reported on some work involving the conversion of scores produced by a template based recogniser into estimates of probability. They derive a formula for this conversion based on assumptions of independence of speech frames and Gaussian distributed distances. Here we use an alternative formulation which makes fewer assumptions but is not a true probability, rather an estimate of likelihood.

During the training phase of reference template selection for each vocabulary word we can calculate a distance matrix representing the similarities between every training example of that word (this is often a byproduct of the clustering algorithm). These distances are considered to represent the likely distribution of scores when a template has been correctly matched against an unknown utterance. If $p_i(x)$ is a probability density function which approximates the true distribution of distances for a correctly recognised word W_i then we can estimate the likelihood that the j'th reference template of word i (W_{ij}) generating a distance d_{ij} is a good match, as

$$l(W_{ij}) = 1 - \int_0^{d_{ij}} p_i(x).dx \qquad (1)$$

The integral above represents the cumulative density function of $p_i(x)$ and will often have no closed form. However, simple non-iterative numerical approximations are available for calculating the tail probability of many cumulative density functions and so we do not have to resort to numerical integration. The distribution of distances produced during clustering have been examined for the data described in the results section. Visual inspection of histograms showed that distribution of distances for each word were approximately Gaussian.

IMPLEMENTATION OF ADC DECISION RULES

For the experiments described here we have used only very simple ADC's. During template training, the reference templates for male and female speakers are constructed separately. This allows an ADC C_{mf} containing only 2 subsets to be used, splitting the reference template inventory I into male and female subsets (M,F). Since C_{mf} is a very powerful ADC, enabling a binary decision on I, the decision rules $R_-(S)$ and $R_+(S)$ must be very strict. Both $R_-(S)$ and $R_+(S)$ are based on a hypothesis test similar to the sign test for paired samples. For $R_-(S)$ we try to reject a null hypothesis that the members of S are performing no better than their competitors in favour of an alternative hypothesis that S is performing significantly worse, at a specified level of confidence (we use 80%). The relative merit of S is assessed by recording the number of times S matches worse than its competitors using a modified K-nearest neighbour (Knn) criterion [1]. This calculates an average likelihood estimate on the K members of S giving highest likelihood for the assumed correct word (word with lowest distance score). If m is the number of times S has matched worse and n is the number of comparisons made over a series of recognition cycles then

$$\text{accept } R_-(S) \text{ if } (1 - B(m,n,½)) > 0.8 \qquad (2)$$

where $B(m,n,½)$ is the cumulative binomial function and the value 0.8 represents an 80% confidence level.

Unfortunately, we cannot use exactly the same test for the positive decision rule $R_+(S)$ since no information about the S under consideration is available. Instead, any decision must be made on the basis of poor performance of the currently active subset $\neg S$ (i.e. if M is active and we are considering $R_+(F)$ then $\neg F=M$). In order to quantify poor performance a β threshold is defined. Values obtained from the Knn calculation on $\neg S$ falling below this threshold are indicators that adaptation has failed. If l is the number of times the β threshold has not been met and n is the number of comparisons made since S became inactive then

$$\text{accept } R_+(S) \text{ if } (1 - B(l,n,½)) > 0.8 \qquad (3)$$

We also define a further group of ADC's which allow individual reference templates to be removed from the active inventory. Since each word W_k has associated with it J reference templates W_{kj} (j:1..J) we use a separate ADC (C_k) for each vocabulary word. Every C_k has J subsets S_j (j:1..J) each containing only a single reference template (W_{kj}), which, of course, may be active or inactive (here the use of sets is purely for consistency). The negative decision rule $R_-(S_j)$ operates in the same manner for each C_k, similarly with $R_+(S_j)$. Since each ADC is quite weak, only one reference template can be made inactive, relatively little information needs to be collected before a decision can be made. For the negative decision rule $R_-(S_j)$, information supporting the hypothesis that any S_j is not effective can only be collected if its member belongs to the class (C_k) associated with the word W_k that has been assumed correctly recognised. Furthermore, some template for W_k must have matched with a likelihood exceeding an α threshold. This is set high enough to ensure that adaptation only takes place when there is a high likelihood that W_k is correct. Subject to these conditions and the constraints on decision rules, S_j can be made inactive only if it has generated the lowest likelihood estimate of any in its class below the β threshold mentioned above. The positive decision rule $R_+(S_j)$ allows the member of S_j to be reactivated if it was the most recently deactivated of the inactive reference templates contained within (C_k) and that the assumed recognised word W_k has a likelihood below the β threshold.

PERFORMANCE RESULTS

In order to evaluate the procedures previously described, reference templates were created for the digits vocabulary (zero, one, .. nine). Fourteen speakers, seven male and seven female, were recorded onto cassette tape repeating each of the digits four times. This data was then digitised at 10kHz using a 12 bit A/D converter. An automatic endpoint detection algorithm was used to isolate the words from their surrounding silence. Pre-emphasis and 15-pole linear prediction analysis were then performed using 256 point frames with a 128 point overlap producing one frame of autocorrelation and linear prediction coefficients every 12.8 ms. Using the data for each word the parameters of Gaussian probability density functions were calculated which provide the basis for the likelihood estimations needed during the adaptation process. The data was then split into two sections containing separately the male and female speech. For each word and each section, 4 reference templates were produced by clustering the data into 4 clusters using the K-means algorithm and averaging the contents of the cluster about the cluster center [6]. After clustering we are left with 8 reference templates per word which hopefully represent the speaker variation present within the contributing male and female speakers.

DYNAMIC SPEAKER ADAPTATION

To evaluate the performance of the recogniser, test data was also collected from two male speakers, not present in the training data, each providing twenty repetitions of the digits vocabulary, spoken in groups of four with other digits interleaved, and digitised using the same procedure as above. The data provided by each test speaker was then presented in random order to an isolated-word DP recogniser [5] using the above reference templates. The results for speaker 1 are summarised in table 1. For each speaker the thresholds used in the adaptation process were set at $\alpha = 0.9$, $\beta = 0.6$ and $K = 3$.

	State after cycles				
	10	20	50	100	200
templates active	47.50%	45.00%	41.25%	40.00%	40.00%
adaptive method	100.00%	100.00%	98.00%	98.00%	97.50%
standard method	100.00%	95.00%	96.00%	94.00%	95.00%

Table 1. (speaker 1)

As can be seen in table 1, after 10 recognition cycles only 47.5% of the original template inventory are still active while no mistakes have been made in recognition. The 'adaptive method' row shows the percentage recognition rate obtained when adaptive methods are used, while the 'standard method' row shows the equivalent situation if all the reference templates are used (percentages are calculated using a moving average). After 50 cycles several confusions which would normally be present if the whole template inventory were used have been eliminated. In the long term case, at 200 cycles, we end up with a 2.5% increase in recognition accuracy while only using 40% of the reference template inventory.

	State after cycles				
	10	20	50	100	200
templates active	48.75%	47.50%	45.00%	41.25%	37.00%
adaptive method	60.00%	75.00%	84.00%	83.00%	85.00%
standard method	60.00%	70.00%	74.00%	78.00%	79.50%

Table 2. (speaker 2)

Table 2 shows that even better improvements can be made for a different speaker who normally has a quite poor recognition rate. After 200 cycles, we end up with a 5.5% increase in recognition accuracy while only using 37% of the reference template inventory. In both cases, the major ADC splitting the template inventory into male and female speakers, has been invoked after five cycles, while the minor ADC's continue to lower the number of active reference templates.

CONCLUSIONS

We have presented a method for adapting the scope of speaker-independent reference templates to suit the current speaker. We have shown that this can both improve recognition accuracy by reducing possible cross-talker confusions and dramatically decrease the size of the template inventory. Furthermore, the method can operate fast enough to be practically useful in a speaker-independent recognition system. However, things can go wrong if many incorrect recognitions are among the first few utterances. In this critical case, adaptation decisions based on incorrect information are made which prejudice further attempts at recovery. The decision rules which attempt to undo incorrect adaptation seem, at the moment, rather too weak for a practically robust system. We are therefore investigating how external feedback from higher level processes can help to make the adaptation process more stable. We are also considering ways in which this kind of adaptation can be modified for use in connected word DP algorithms and how to extend the knowledge used by ADC's to make adaptation faster.

ACKNOWLEDGEMENTS

This work was supported by a joint SERC/British Telecom grant (BT/SERC Contract No. SR/D/4361.7). The speech data used to create speaker-independent reference templates was collected by British Telecom. Acknowledgement is made to the Director of Research at British Telecom for permission to publish this paper.

REFERENCES

[1] L.R. Rabiner, S.E. Levinson, A.E. Rosenberg and J.G. Wilpon, "Speaker-Independent Recognition of Isolated Words Using Clustering Techniques," IEEE Trans. Acoust., Speech., Signal Processing., Vol. ASSP-27, pp. 336-349, Aug 1979.

[2] L.R. Rabiner, S.E. Levinson and M.M. Sondhi, "On the Application of Vector Quantisation and Hidden Markov Models to Speaker-Independent, Isolated Word Recognition," Bell System Tech. J., Vol. 62, pp. 1075-1105, 1983.

[3] T.R.G. Green, S.J. Payne, D.L. Morrison and A. Shav, "Friendly Interfacing to Simple Speech Recognisers," Behaviour & Information Technology., Vol.2, No.1, pp. 23-38, 1983.

[4] R.I. Damper and S.L. MacDonald, "Template Adaption in Speech Recognition," Proc. Inst. Acoustics., Vol.6, Part.4, pp. 293-299, 1984.

[5] H. Sakoe and S. Chiba, "Dynamic Programming Algorithm Optimisation for Spoken Word Recognition," IEEE Trans. Acoust., Speech., Signal Processing., Vol. ASSP-26, pp. 43-49, Feb 1978.

[6] J.G. Wilpon and L.R. Rabiner, "A Modified K-Means Clustering Algorithm For Use In Isolated Word Recognition," IEEE Trans. Acoust., Speech., Signal Processing., Vol. ASSP-33, pp. 587-594, Jun 1985.

[7] C.J. Clotworthy and F.J. Smith, "Spoken Word Variation and Probability Estimation," IEE Conf. Speech IO., pp. 27-30, 1986.