ISCA Archive
http://www.isca-speech.org/archive

European Conference on
Speech Technology
Edinburgh, Scotland, UK
September 1987

USING CONDENSED NEAREST NEIGHBOR RULE FOR SPEAKER INDEPENDENT WORD RECOGNITION

N. Yalabik*, A. Mansur*

ABSTRACT

In this study, a combination of Editing and modified Condensed Nearest Neighbor (CNN) rule is used for classification of speaker independent isolated words. The approach is compared to clustering techniques used for template selection. The implementation is done with an 18-word Turkish vocabulary where Linear Preditive Coding parameters and Dynamic Time Warping are used for classification. The results show that Editing and CNN together can be used for template selection as a better alternative to clustering.

INTRODUCTION

Word recognition systems usually treat words as a whole and the classification is based on the similarity of the unknown sample to class-representative samples, called templates. Speaker independent recognition requires the use of at least a few templates per word, to capture the inherent variability of speech among speakers. Commonly, these samples are chosen using various clustering techniques (Ref 1). Clustering makes use of the assumption that the samples from a word tend to form a few natural clusters. However, the clustering does not make any use of the between-category distances and relative position of samples from different categories.

In this paper, we propose an alternative approach for template selection, namely, editing and condensing of the learning samples. Conceptually, editing eliminates samples that fall in Bayes rejection regions. Condensing selects the samples near decision boundaries. The following sections describe these methods and their application to a multicategory, multidimensional word recognition problem, after a brief summary of the recognition system used.

WORD RECOGNITION SYSTEM

The word recognition system used here has been developed earlier and discussed in Ref 2 in detail. Digitized speech signal first goes through end-point detection and preemphasis filtering stages. Then, it is divided into overlapping frames 40ms long, and for each frame, the first 9 Linear Predictive Coding parameters are found. The distance between two words is obtained by using a Dynamic Time Warping algorithm between the LPC coefficient sequences of the test and reference words. The distance between two frames is found using Itakura distance measure. For classification, the Nearest Neighbor rule is applied to selected samples.

The learning phase consists of the collection of a large number of samples from people of different sex, age, etc. and selection of templates from these. In the original study, the minimax clustering algorithm is used for this purpose. The experiments showed that the classification rate improved significantly with an increase in the number of templates used per word, and about 87% of correct recognition was obtained when more than 3 templates per category were used.

* Drexel University, Electrical and Computer Engineering Department, Philadelphia, PA 19104

10.21437/ECST.1987-41

EDITING AND CONDENSING FOR NEAREST NEIGHBOR CLASSIFICATION

NN classification rule was shown to perform very well on the average if the number of learning samples used in the classification is quite large. However, the computational complexity of the algorithm makes it unfeasible especially for multi-category, multi-dimensional problems.

To reduce the number of samples without degrading the system performance, a number of algorithms are suggested. The multiedit algorithm (Ref 3) performs an iterative elimination by dividing the learning samples into groups, using one group to classify the other and eliminating the misclassified samples. It was shown (Ref 3) that the multiedit algorithm improves the recognition rate since "outcasts" or "bad samples" have a good chance of being eliminated.

To reduce the number of samples further, condensing is employed (Ref 4). Condensing basically uses the idea of obtaining a subset of a given set such that the selected subset classifies all the rest correctly. Condensed Nearest Neighbor (CNN) rule can be summarized as follows:

1. Set two bins called S_o and S_n, by putting all samples initially to S_n.
2. Move samples from S_n to S_o one by one if they are incorrectly classified by the samples that are already in S_n.
3. Iterate (2) until no samples can be transferred from S_n to S_o.
4. Use the samples in S_o for classification.

It can be observed that the condensing algorithm can result in different templates depending on how the samples are ordered in S_n. As a remedy, some initial ordering algorithms were proposed (Ref 5, Ref 6). In these, the samples are ordered in a way that the ones close to decision boundaries come first. This provides more chance for the samples near decision boundaries to be selected as templates. These algorithms are generally called modified CNN.

EDITING

Multiedit algorithm requires a very large sample set in order to be effective, which might not be practical, as in our speech recognition problem here. In this study, the following algorithm is used to obtain a similar effect with relatively smaller sample sets:

1. Find a cluster center for each categorgy, and call the collection of all samples that are at a certain range to their own cluster centers S_c. The cluster center is defined as one of the samples such that the average distance from it to all other samples (in the same category) is minumum. The range is chosen seperately for each category to include at least half of the total samples in that category.

2. Use S_c to classify all the rest using the NN rule. Call correctly classified samples S_{cor}.

3. Use S_c U S_{cor} as the edited set as an input to CNN.

MULTICATEGORY MODIFIED CNN

Modified CNN for a multicategory problem becomes computationally prohibitive especially when the number of categories is large, as in the case here, since all distances between all pairs of samples should be calculated. In this study, we used the following algorithm, which applies a separate modified CNN to selected groups of close packed cluster, with a final overall cleaning, as follows:

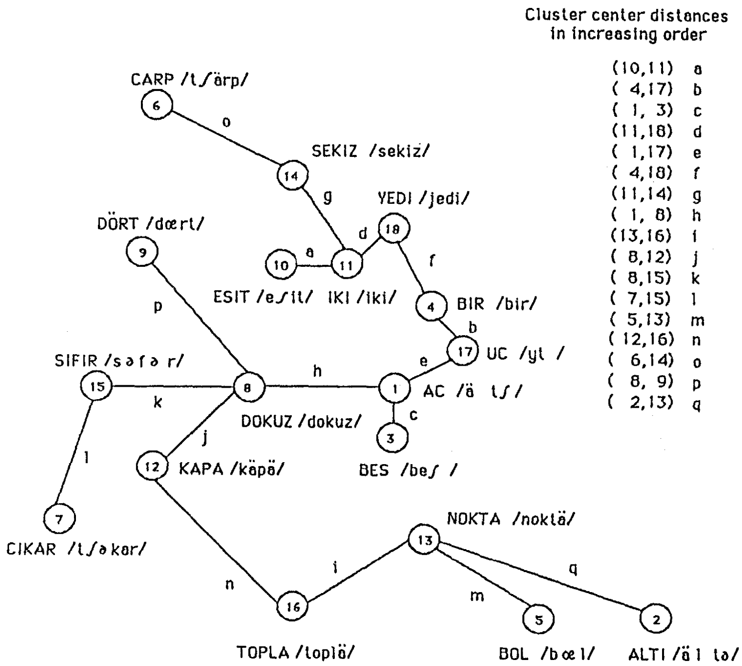

Fig. 1 Minimum Spanning Tree for Cluster Centers of 18 Turkish words

1. Generate a "minimum spanning tree" of the category cluster centers, as shown in Figure 1.

2. For each category, apply modified CNN to it and all others that are connected to it by an edge. For example, UC, AC and BIR in Figure 1. Repeat for all categories. (As the modified CNN, the algorithm in Ref 5 is used.)

3. Find a union of all samples selected in 2, call it S_{un}.

4. Use S_{un} to classify the eliminated samples. If a sample is misclassified, add it to S_{un}. Iterate until all samples are tested.

5. If one complete pass is made through 4 with no transfer to S_{un} then exit or else go to 4.

Experimental Results

In the experiments, 333 samples collecIted from 6 men, 3 women are used as the sample set. 100 samples that are collected from 3 men and 3 women are used for testing. The resultls can be summarized as shown in table 1.

	average number of samples/word	recognition rate
Editing + CNN	6	91
CNN only	8	77
Clustering	6	85

Table 1. Classification results

CONCLUSIONS

As can be observed in table 1, Editing + CNN noticably improved the classification rate compared to clustering. As can be seen, CNN without editing does not give good results, with the reason that it is highly sensitive to "bad" samples.

Using minimal spanning tree concept in multicategory CNN canbe improved by considering some more pairs, which are close but are not included in the tree.

ACKNOWLEDGEMENTS

The authors would like to thank Dr. Banu Onaral for her cooperation in the use of laboratory facilities.

REFERENCES

1. L.R. Rabiner, S.E. Levinson, A.E. Rosenberg, J.G. Wilpon, IEEE Trans. ASSP V27, 336 (1979).
2. N. Yalabik, F. Unal, NATO ASI Series F16 (R.D. Mori and C.Y. Suen eds.), 419, (1985).
3. P.A. Devijver, J. Kittler, Proc Fifth Int. Conf. PR, 72 (1980).
4. P.E. Hart, IEEE Trans. IT V14, 515 (1968).
5. K.C. Gowda, G. Krishna, IEEE Trans IT V25, 488 (1979)
6. I. Tomek, IEEE Trans. SMC, 769 (1976).