EECS225d Audio Signal Processing in Humans and Machines Spring 1995

Lecture 32: April 10, 1995

Lecturer: Prof. Nelson Morgan *Scribe: Warner Warren*

Pattern Classification and Deterministic Sequence Recognition

32.1 Overview

The lecture starts with a discussion of some practical considerations for the training and evaluation of a speech recognizer. Then it moves on to discuss some techniques for isolated word recognition (with an emphasis on dynamic programming), and it concludes by discussing ways to extend these techniques to the connected speech recognition problem. It emphasizes The Dynamic Time Warp algorithm, The K-means method and the one and two pass Time Warp procedures for continuous speech recognition.

32.2 Pattern Classification Techniques

- Feature vector extraction
- Deterministic pattern classification (distances, discriminant functions)
- Statistical pattern classification (approx to Bayes classifier)
  - Gaussian (univariate, multivariate)
  - Simplified forms given covariance assumptions (e.g., linear discriminant)
  - Gaussian mixture (use the EM algorithm)
  - Discrete Distributions
  - Kernel estimators: Probabilities computed directly from data. They are locally smooth.
  - Neural Networks: Estimate posterior probabilities

32.3 Practical Training Issues

The following are some of the important considerations for the design and evaluation of a speech recognition system:

- Feature vector Dimension, discrimination
- Training set size, test set size
- Number of parameters and overfitting (stopping criterion): If the network has too many parameters it can overfit the data if it is trained too long. That is it will memorize the training data, and perform poorly on the test data. Hence, there should be a stopping criterion for training.

- Jackknife procedures: Partition the data into N groups use the first N-1 groups for training, and the Nth group for testing. Re-partition the data, and repeat the procedure. Repeat this this procedure to estimate recognition statistics.
- Testing Significance: Many techniques exists for evaluating the statistical significance of experimental results, but in general, the more testing data one uses the more reliable the results. As with any experiment, the significance should be reported along with the results. The following is a simple Significance test:
  1. Model variation between the methods as a random coin toss for correct/incorrect binomial distribution with prob(correct) = .5
  2. For enough coin tosses, resulting distribution is approximately Normal with a SD of $.5n^{1/2}$, where n is the number of test patterns.

     For example, in order to have 99% confidence that the result is genuine (.01 level), we need 2.33 SDs or

     $Gen.Dif. = 100 * 2.33 * .5n^{1/2}/n = 117/n^{1/2}$
- Testing on the Training set: Avoid testing on the training set since this will tend to give recognition rates which are artificially high. Obviously, recognition data obtained in this way tell little or nothing about how the actual system performs on independent test and training sets.
- Training on the Test set: Also devlopment is typically done on the test set, so that design decisions may be somewhat tuned to that test set. In other words, during development, changes are made to system to improve its performance on the test set. However, these changes may not improve performance on other data. In order to fairly evaluate the affects of these changes a third evaluation set, which was neither used for testing or training, should be used.

32.4 Sequence Recognition

Automatic speech recognition systems combine pattern recognition techniques, which identify subword units, with sequence recognition techniques, which identify sequences of subword units that make up words and sequences of words that make up connected speech. The following sequence recognition techniques are considered:

- Template Matching
- DTW
- HMMs
- Hybrid NN/HMM

32.4.1 Speech Unit Models

For ASR, speech is represented as either word templates or Stochastic Models. This lecture focuses primarily on word template models with discussion of sequence matching.

32.4.2 Word template models

Each word is represented by one or several of its pronunciations. No training per se is required[1]. Recognition is done with a comparison test of the utterance against a set of reference patterns to find the closest one.

But: since there is deformation along the time and frequency axes, how do we define the closest pattern?

32.4.3 Cost Estimation

In order to perform template matching one must first decide upon a cost function, i.e. an objective function to be minimized by the optimization procedure. Once the cost function is selected (Euclidean distance, Spectral distance, etc.), a dynamic programming algorithm is used the find the reference pattern which minimizes the cost function for a given input pattern. Local Cost are computed on a frame by frame basis within words. The optimization procedure also performs time alignment between the the input frames and the reference frames to mitigate the effects of variation in speaking rates.

32.4.4 Time Registration

These are some of the methods of performing time alignment between the reference word and the input word.

Linear Alignment: This is a simple linear warping (stretching or contraction) of the time axis. Although it sometimes works, it is generally inferior to methods such as DTW. One reason for this is the fact that some sounds in the input word may be speeded up (or slowed down) more than others, relative to corresponding sounds of the reference word.

Time event Matching ane Correlation Matching: These method align data by finding the times at which the input signal matches the reference signal.

Dynamic Time Warping (DTW): This is a dynamic programming technique used to create a nonlinear warping function between the input time axis and the reference time axis.

[1] Although there is no explicite parameter adjustment phase, the process of aquiring and storing the templates might be considered to be form of training

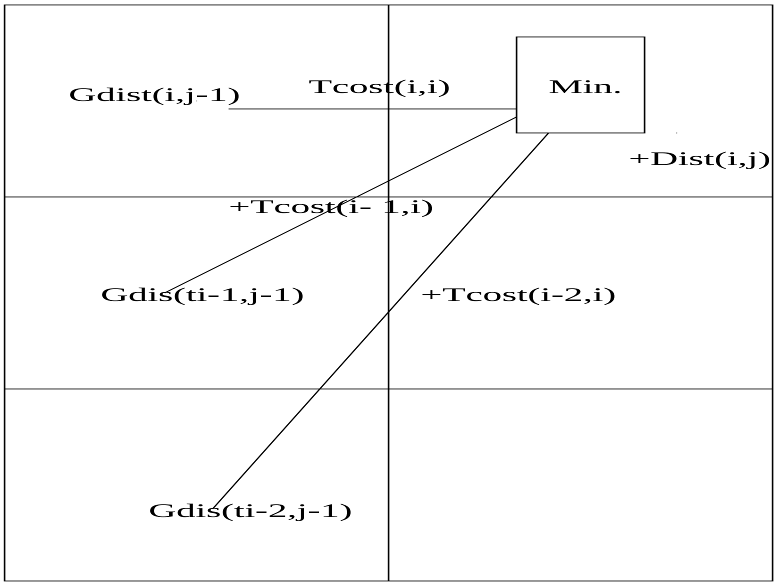


Figure 1 Dynamic Programming Example

32.4.5 Dynamic Programming

- Bellman Optimality Principle: if the best path goes through a grid point, then the best path includes the best partial path to that grid point.
- Define a local and global distance
- Simple recursion to get global distance

$$GDist_{i,j} = Dist_{i,j} + min_k[GDist_{k,j-1} + Tcost_{k,i}]$$

Where k runs over all possible predecessors of i, and $d(X_R, Y_T) = GDist_{R,T}$. This computation is illustrated in figure 1. The computation typically is done by running up each column, starting with the leftmost.

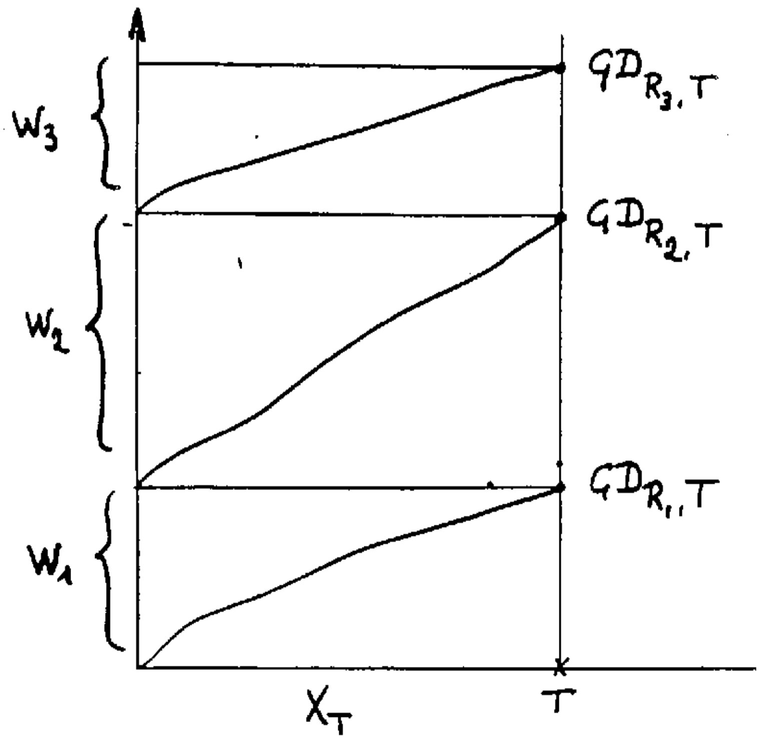


Figure 2: An isolated word recognition example

32.5 Overall System

The overall recognition system comprises training (acquisition), clustering and testing. While word template models don't require any training, they require a lot of computation and they only model words. Grouping reference vectors into clusters (instead of storing all spoken examples) reduces the number of comparisons that must be performed. This reduces the amount of computation required for template matching.

32.5.1 Clustering

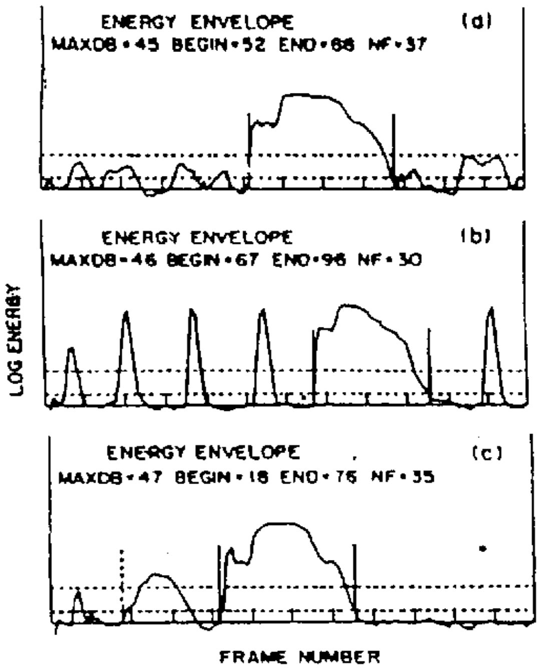
Three methods of clustering are considered. We can do no clustering at all. We can use the use the DTW for alignment and then average over all examples of an utterance, or we can use the K-means technique to do true clustering.

32.5.2 The K-means Technique

1. Choose K cluster centers at random
2. Assign every y_j to cluster k if: $d(c_k, y_j) < d(c_i, y_j)$, all i not equal to j
3. update $c_k' s$
4. select a stopping criterion, return to step 2, if not done.

During recognition, $d(x_i, y_j)$ is replaced by $d(c_k, y_j)$. This implies that there are a maximum of K distances to compute. Furthermore, the procedure produces a smoothing effect. This method also yields the best results with the least computation.

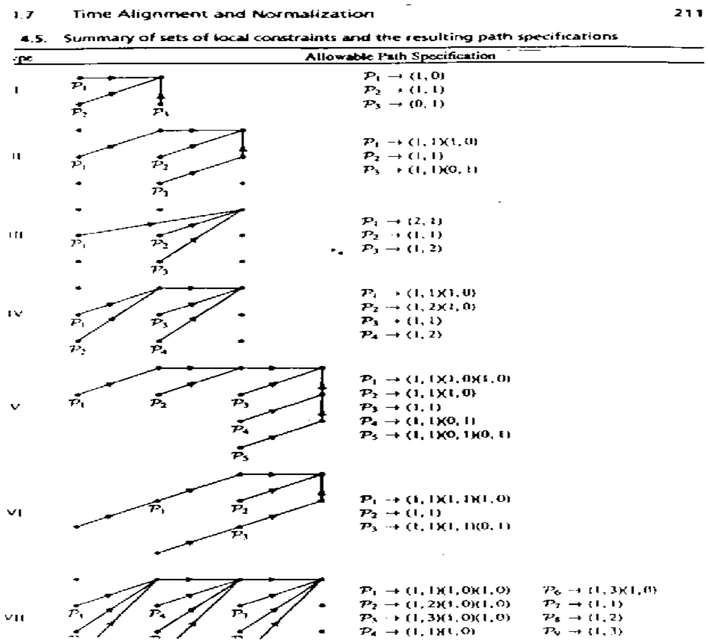
32.6 DTW Requirements



Three cases of endpoint detection using energy contours.

Figure 3. End point detection

- End point constraints: End point detection caused most problems for isolated word recognition systems.
- Local path constraints: Local path constraints specify how paths may enter a node. Below is an example of the the local path constraints used in a DTW system from AT&T:
- Global path constraints: These constraints specify the allowable regions for the warping functions.
- Distance Measure: Three frame-frame distance measures used with the DTW are:
  - Euclidean Distance
    $\| X_i - X_r \|$
  - Covariance weighted (Mahalanopis) Distance
    $(X_i - X_r)^T \sum (X_i - X_r)$
  - Spectral Distance
    $\int |(A_i\omega) - A_r(\omega)|^2 d\omega$
  - LPC log likelihood (Itakura)
    $\int \log \frac{|A_i(\omega)|^2}{|A_r(\omega)|^2} d\omega$



1.7 Time Alignment and Normalization 211

4.5. Summary of sets of local constraints and the resulting path specifications

Type	Allowable Path Specification
I	$\mathcal{P}_1 \rightarrow (1,0)$ $\mathcal{P}_2 \rightarrow (1,1)$ $\mathcal{P}_3 \rightarrow (0,1)$
II	$\mathcal{P}_1 \rightarrow (1,1)(1,0)$ $\mathcal{P}_2 \rightarrow (1,1)$ $\mathcal{P}_3 \rightarrow (1,1)(0,1)$
III	$\mathcal{P}_1 \rightarrow (2,1)$ $\mathcal{P}_2 \rightarrow (1,1)$ $\mathcal{P}_3 \rightarrow (1,2)$
IV	$\mathcal{P}_1 \rightarrow (1,1)(1,0)$ $\mathcal{P}_2 \rightarrow (1,2)(1,0)$ $\mathcal{P}_3 \rightarrow (1,1)$ $\mathcal{P}_4 \rightarrow (1,2)$
V	$\mathcal{P}_1 \rightarrow (1,1)(1,0)(1,0)$ $\mathcal{P}_2 \rightarrow (1,1)(1,0)$ $\mathcal{P}_3 \rightarrow (1,1)$ $\mathcal{P}_4 \rightarrow (1,1)(0,1)$ $\mathcal{P}_5 \rightarrow (1,1)(0,1)(0,1)$
VI	$\mathcal{P}_1 \rightarrow (1,1)(1,1)(1,0)$ $\mathcal{P}_2 \rightarrow (1,1)$ $\mathcal{P}_3 \rightarrow (1,1)(1,1)(0,1)$
VII	$\mathcal{P}_1 \rightarrow (1,1)(1,0)(1,0)$ $\mathcal{P}_2 \rightarrow (1,2)(1,0)(1,0)$ $\mathcal{P}_3 \rightarrow (1,3)(1,0)(1,0)$ $\mathcal{P}_4 \rightarrow (1,1)(1,0)$ $\mathcal{P}_6 \rightarrow (1,3)(1,0)$ $\mathcal{P}_7 \rightarrow (1,1)$ $\mathcal{P}_8 \rightarrow (1,2)$ $\mathcal{P}_9 \rightarrow (1,3)$

Figure 4. Local path constraints used in an AT&T DTW system

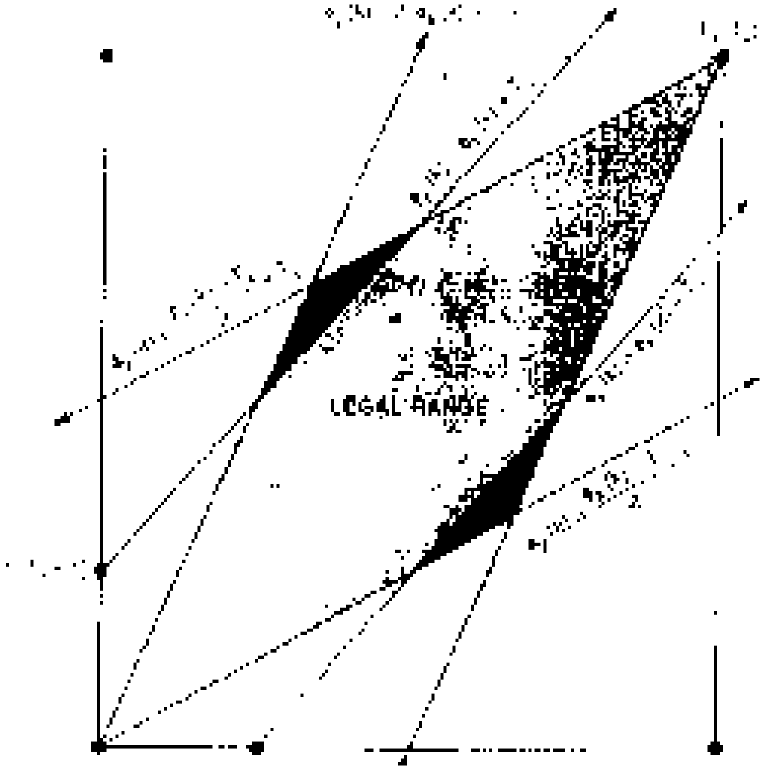


Figure 5. Global distance constraints

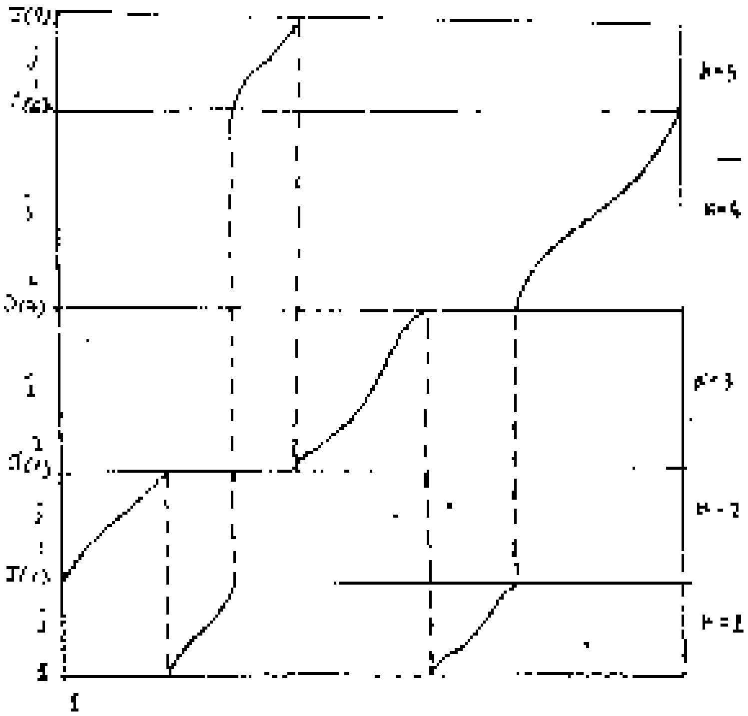
32.7 Time Warp for Continuous Speech

The time warping technique is also used to in continuous speech recognition systems. Within each word, the isolated word dynamic programming technique is used. Between words, the cost of going from the end of one word to the begining of the next word is minimized. The basic procedure is:

- Inside each reference word use the Dynamic Programming procedure
- At the first frame of each reference word compute:

  $GD_{1,j} = min_k[GD_{k,j-1} + Tcost_{k,1}]$

  Where k = i, last frame of each possible previous words.



Time frames of input pattern for Connected Word Time Warp

Figure 6. Sequence Time Warping

The following techniques are discussed:

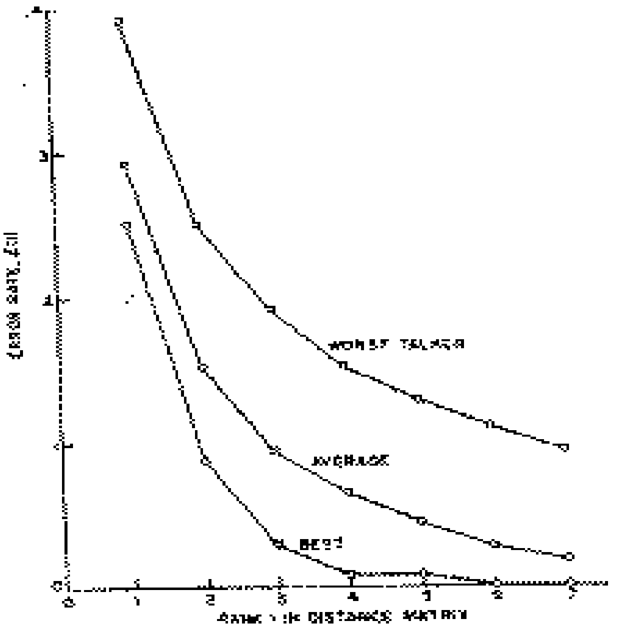
- Sakoe 2-pass method
- Myers and Rabiner level-building
- One-Pass method: Vintsyuk, recalled by Bridle, tutorial by Ney [1]

32.7.1 One Pass Method

- Same as isolated word recognition within word; consider possible transitions across words
- Reduce storage with backtracking vectors
- Syntax easily added
- Reduced computation versus multi-pass approaches

32.8 Concluding Remarks

The Sheep and Goats analogy. The results obtained using these methods tend to show that the best scores tend to be very close to the average score while the worst scores tend to be very far from the average.



Recognition error rate as a function of rank in the distance matrix for the worst, average, and best talkers.

Figure 7. Sheep and Goats

32.9 References

[1] H. Ney, "The use of a One Stage Dynamic Programmming Algorithm for Connected Word Recognition," IEE Trans. Acoust., Speech, Signal Processing, , Vol. ASSP.32, No. 2, pp. 263-271,April 1984.