BRIEF ARTICLE

THE AUTHOR

1. Motivation

In the previous chapter, we have seen that using a MFCC representation of utterances with regions of silence removed leads to a large improvement in accuracy, time and computational complexity in the performance of DTW algorithm augmented with a euclidean metric..The main contributing factor behind the large time and computational complexity of base line DTW is the **size** of the time series sequences. The computational cost of a DTW algorithm is (mn) where m and n denote the length of the two time series sequences currently compared. Using longer sequences thus increases the size of the DTW cost matrix hence resulting into a greater number of computations.

The DTW algorithm on its own is a domain independent algorithm that uses a similarity metric to implicitly extract information about global trends. The algorithm employes dynamic programming to search a space of mapping between the time axis of the two respective sequences to determine the optimum alignment between them. The only difference between MFCC-augmented DTW and baseline DTW is the feature extraction stage. In machine learning, feature extraction refers to the pre-processing stage that involves the extraction of new features from a set of raw attributes through a suitable functional mapping. The extraction phase of MFCC features involves a segmentation of the time series followed by a functional mapping on the segmented windows. The resultant sequence of extracted feature vectors has a much smaller length compared to the length of the original sequence. Evident from the experiments done in the previous chapters, it can concluded that the use mel-cepstrum features based on 'cleaned' signals not only increases the accuracy of DTW but also reduces the time and computational cost through reduction of dimensionality of the original sequence.

In this section, to tackle the problems of accuracy, time and computational complexity faced by the base line DTW, I investigate a self-proposed data-driven methodology that can be partitioned in the following two stages:

- Feature extraction
- Kernel construction

The feature extraction stage is motivated from the extraction of MFCCs. The utterances are segmented into windows of width 10 ms and functional mapping is applied to each window. The mapping that I chose in this particular instance is as follows:

Stage 1 only increases the dimension of each point of time. The feature extraction stage does do anything to reduces the size of the time series sequence.. In the 2nd stage, I propose a kernel that computes similarity using segmented

Limitation of value-based DTW :

- Ignores the context such as their positions in local features and their relations to overall trends.

Limitation of derivative based DTW :

- Fails to detect significant common sub-patterns between two sequences(mainly global trends)

Thus we need an algorithm that not gains vision over overall shapes but also on local trends. ****

Definition of local feature:

$$f_{\text{local}}(r_i) = (r_i - r_{i-1}, r_i - r_{i+1})$$

Definition of global feature: Points to consider: must reflect information about the global trends and in order to be combined with local features, they must be of the same scale.

$$f_{\text{global}}(r_i) = (r_i - \sum_{k=1}^{i-1} r_k, r_i - \sum_{k=i+1}^{M} \frac{r_k}{M+1})$$

Kernel functions must be continuous, symmetric, and most preferably should have a positive (semi-) definite Gram matrix. Kernels which are said to satisfy the Mercer's theorem are positive semi-definite, meaning their kernel matrices have no non-negative Eigen values. The use of a positive definite kernel insures that the optimization problem will be convex and solution will be unique.

$$\begin{aligned} k(x,z) &= (x^T x')^2 \\ &= (x_1 z_1 + x_2 z_2)^2 \\ &= x_1^2 z_1^2 + 2x_1 z_1 x_2 z_2 + x_2^2 z_2^2 \\ &= (x_1^2, 2x_1 x_2, x_2^2)(z_1^2, 2z_1 z_2, z_2^2)^T \\ &= \phi(x)^T \phi(z) \end{aligned}$$

We saw that the simple polynomial kernel $k(x,z) = (x^T z)^2$ contains only terms of degree two. If we consider the slightly generalised kernel:

$$(x,z) = (x^T z + c)^2$$

with c >0, then the corresponding feature mapping $\phi(x)$ contains constant and linear terms as well as terms of order two. If we generalize this notion then $k(x,x') = (x^T z)^M$ contains all monomials of order M. For instance, if x and z are two images, then the kernel represents a particular weighted sum of all possible products of M pixels in the first image with M pixels in the second image.