Clustering Speech Samples based on Relative Distance Measure

Sunil Kumar Kopparapu and Meghna A Pandharipande

TCS Innovation lab - Mumbai,
Tata Consultancy Services Limited,
Yantra Park, Thane (West) - 400 601.
{sunilkumar.kopparapu, meghna.pandharipande}@tcs.com

ABSTRACT

Clustering of data is a well addressed problem in several fields and finds use in various applications. Almost all the clustering techniques work on a data-set where each element in the data-set is represented by a set of parameters which are defined with respect to some reference point. Clustering of speech samples is important in several applications. In this paper, we propose a novel clustering technique that aids in clustering speech data. One of the major differences between the clustering technique proposed in this paper and the existing ones is in clustering the data based on *relative* distances between the speech samples and not the absolute distances from a common reference point as used in other clustering algorithms. We initially show the performance of the proposed clustering algorithm on some standard data-sets and follow it up with real speech data-set clustering.

Keywords: Clustering, Speech Signal, Dynamic Time Warping, HMM

1. MOTIVATION

Several clustering algorithms exist in literature [1, 2, 3, 4, 5] and more recently [6] captures the literature under one roof. Almost all of them work on the assumption that the data-set to be clustered contains data points that are represented by a vector that has a common reference point which is more often a pre-defined reference origin. More formally, conventional clustering algorithms work on a set of objects $\mathscr{S}$ which are represented by a feature vector with reference to some origin. For example, say there are N objects $\mathscr{S} = \{s_1, s_2, \cdots, s_N\}$ which are to be clustered into m ($< N$) classes. Let each object s_i be represented by, a feature vector, F, of dimension k, namely $F_i = (f_{i1}, f_{i2}, \cdots, f_{ik})$. Notice that the feature vector F_i representing the object s_i is with reference to *a common* origin $\mathscr{O}$, typically, $\mathscr{O} = (o_1 = 0, o_2 = 0, \cdots, o_k = 0)$ and the dimension of the feature vector of all the objects is same, namely, k. Almost all clustering techniques use some form of metric (say $\mathbb{L}^2$ norm) to compute the closeness of two objects to group them together.

Conventional methods of clustering are not applicable to cluster speech signals because the feature vectors extracted from a speech signal are typically of unequal length because the length of the speech signal corresponding to the same spoken word is more often of different length. Typical cause of this are (a) people have different speaking rate and also the same person under different contexts speaks at a different rate, (b) the same utterance is spoken in different accent by different people and more commonly (c) different vowels are stretched differently by different speakers. Hence, the feature vectors representing speech signals is of varying length. In this scenario, the metrics used by conventional clustering algorithms can not be used for speech signals. In speech literature, dynamic time warping (DTW) [7] and hidden Markov models are typically used to *compare* two speech signals represented by different length feature vectors.

In signal processing literature, the Linde-Buzo-Gray (LBG) algorithm [8, 9] based on Itakura-Saito distance, which is a Bregman divergence, has been used by for clustering speech data. An application where speech data also forms dense clusters is voice recognition for biometrics applications [10]. Jin et al [11] discuss an automatic method to cluster speakers. They show experimentally that their proposed speaker clustering algorithm improves unsupervised adaptation as much as the hand labeled ideal case where the clusters are generated based on true speaker, channel and background condition. For clustering speech samples the only information that is available is the relative distances of one speech sample from another speech sample for the purpose of clustering. In this paper we propose an algorithm to cluster speech samples using *only* the relative distances between each other. This aspect motivates the need for a clustering scheme proposed in this paper. The clustering scheme presented here is based on only the relative distances between the objects (or speech samples) to be clustered and not on the absolute distance of the object from a common origin. The relative distances are computed by computing the DTW distance between the two speech signals/samples. The rest of the paper is organized as follow, we introduce some basics on how the features are computed for speech signals

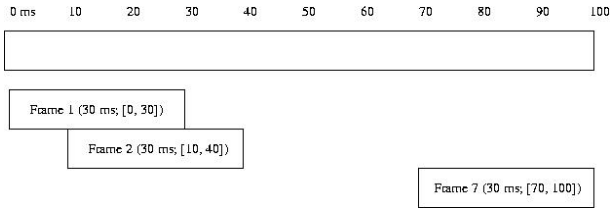

Fig. 1. If Speech signal duration is T (= 100 ms), then total total number of frames (P) is given by (T - F_{size})/F_{shift} = (100 - 30)/10 = 7, provided frame size (F_{size}): 30 ms and frame shift (F_{shift}): 10 ms. Typically, $F_{shift} = F_{size}/3$, $F_{size}/2$

in Section 2 followed by the clustering scheme in Section 3 followed by experimental results in Section 4 and conclude in Section 5.

2. SPEECH FEATURE EXTRACTION

The most commonly used speech features are MFCC (Mel frequency Cepstral coefficients). We show in this section how these speech features are extracted. The extracted speech features feed the clustering scheme to form clusters.

As is common is speech signal processing, for each speech sample $x[n]$, we divided the speech signal into a frame of length 30 ms ($N = 240$ samples if the speech is assumed be sampled at 8 kHz) with an overlap of 15 ms (as shown in Figure 1) resulting in say, P frames such that

$$\left\{\vec{x}_1[n], \vec{x}_2[n] \cdots \vec{x}_p[n] \cdots \vec{x}_P[n]\right\}$$

where $\vec{x}_p[n]$ denotes the p^{th} frame of the speech signal $x[n]$ and is

$$\vec{x}_p[n] = \left\{ x\left[p\left(\frac{N}{2} - 1\right) + i\right]\right\}_{i=0}^{N-1} \tag{1}$$

Now the speech signal $x[n]$ can be represented in matrix notation as

$$\hat{X} \stackrel{def}{=} [\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_p, \cdots, \vec{x}_P]$$

where

$$\vec{x}_p = \begin{bmatrix} x\left[(p-1) * \frac{N}{2}\right] \\ x\left[(p-1) * \frac{N}{2} + 1\right] \\ \vdots \\ x\left[(p-1) * \frac{N}{2} + N - 1\right] \end{bmatrix}$$

Note that the size of the matrix $\hat{X}$ is $N \times P$ and P varies for different speech samples for reasons discussed in Section 1. We extract Mel frequency Cepstral coefficients (speech parameters) for each of these P frames of speech.

In speech signal processing, in order to compute the MFCCs of the p^{th} frame, $\vec{x}_p$ is multiplied with a Hamming window (see Figure 2)

$$w[n] = 0.54 - 0.46 \cos\left(\frac{n\pi}{N}\right)$$

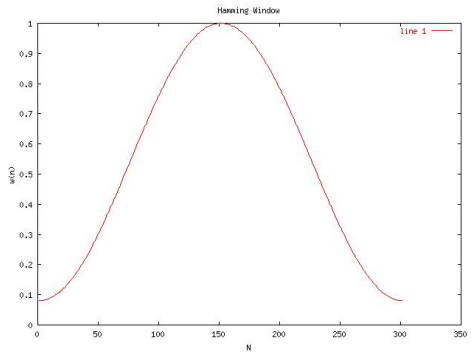

Fig. 2. Hamming Window

followed by the discrete Fourier transform (DFT) as shown in (2).

$$X_p(k) = \sum_{n=0}^{N-1} x_p[n] w[n] \exp^{-j\frac{2\pi kn}{N}} \tag{2}$$

for $k = 0, 1, \cdots, N-1$. If f_s is the sampling rate of the speech signal $x[n]$ then k corresponds to the frequency $f(k) = kf_s/N$.

Let $\vec{X}_p = [X_p(0),\ X_p(1),\ \cdots,\ X_p(N-1)]^T$ represent the DFT of $\vec{x}_p$, then, $X = [\vec{X}_1, \vec{X}_2, \cdots \vec{X}_p, \cdots, \vec{X}_P]$ represents the DFT of the windowed p^{th} frame of the speech signal $x[n]$. Note that the size of X is $N \times P$ and is known as STFT (short time Fourier transform) matrix.

The modulus of Fourier transform is extracted and the magnitude spectrum is obtained as $|X|$ which is a matrix of size $N \times P$. The magnitude spectrum is warped according to the Mel scale in order to adapt the frequency resolution to the properties of the human ear [12]. It should be noted that the relation between the Mel frequency and the linear frequency is given by

$$m_f = 2595 \log\left(1 + \frac{f}{700}\right)$$

where m_f is the Mel frequency and f is the linear frequency [13]. The inverse relationship between f and m_f is given by

$$f = 700\left(\exp^{\left(\frac{m_f}{2595}\right)} - 1\right)$$

Then the magnitude spectrum $|X|$ is segmented into a number of critical bands by means of a Mel filter bank which typically consists of a series of, say, F overlapping triangular filters defined by their center frequencies $f_c(m)$. The parameters that define a Mel filter bank are (a) number of Mel filters, (b) minimum frequency and (c) maximum frequency. For speech, in general, it is suggested in [14] that the minimum frequency be greater than 100 Hz. Furthermore, by setting the minimum frequency above 50/60Hz, we get rid of the hum resulting from the AC power, if present. [14] also suggests that the maximum frequency be less than the Nyquist frequency. Furthermore, there is not much information above 6800 Hz.

The Mel filter bank, $M(m,k)$ [15] is given by $M(m,k)$ $M(m,k)$

$$= \begin{cases} 0 & \text{for } f_k < f_c(m-1) \\ \frac{f_k - f_c(m-1)}{f_c(m) - f_c(m) - f_c(m-1)} & \text{for } f_c(m-1) \le f(k) < f_c(m) \\ \frac{f_k - f_c(m+1)}{f_c(m) - f_c(m) - f_c(m+1)} & \text{for } f_c(m) \le f(k) < f_c(m+1) \\ 0 & \text{for } f_k \ge f_c(m+1) \end{cases}$$

The Mel filter bank $M(m,k)$ is an $F \times N$ matrix. The logarithm of the filter bank outputs (Mel spectrum) is given in (3).

$$L(m,p) = ln\left\{\sum_{k=0}^{N-1} M(m,k)|X(k,p)|\right\} \tag{3}$$

where $m = 1,2,\cdots,F$ and $p = 1,2,\cdots,P$. The filter bank output, which is the product of the Mel filter bank, M and the magnitude spectrum, $|X|$ is a $F \times P$ matrix. A discrete cosine transform of $L(m,p)$ results in the MFCC vector.

$$D(r,p) = \sum_{m=1}^{F} L(m,p)\cos\left\{\frac{r(2m-1)\pi}{2F}\right\} \tag{4}$$

where $r = 1,2,\cdots,F$ and $D(r,p)$ is the r^{th} MFCC of the p^{th} frame. The MFCC of all the P frames of the speech signal are obtained as a matrix Φ

$$\Phi\{\hat{X}\} = D = [\Phi_1, \Phi_2, \cdots, \Phi_p, \cdots \Phi_P] \tag{5}$$

Note that the p^{th} column of the matrix Φ represents the MFCC of the speech signal, $x[n]$, corresponding to the p^{th} frame. Note that for different speech samples the number of frames is different, hence the size of $\Phi_P]$ is different. In a similar fashion Δ MFCC and Δ^2 MFCC are also calculated for all the P frames of a speech sample.

3. METHOD FOR FORMING CLUSTERS

Consider a collection of N speech samples which needs to be clustered into n clusters. Note that these speech samples might correspond to the same spoken word spoken by different people with different accents and different speaking rates. Nevertheless, the dimension of the feature vector F extracted (MFCC, Δ MFCC and Δ^2 MFCC) from each speech sample is different[1] even if the speech sample corresponds to the same spoken word. Let the feature vector of the i^{th} speech sample be represented by F_i which is of dimension p_i. The relative distance between all the N speech samples is computed, namely,

$$\{d_{S_iS_j}\}_{i=0,j=0,j\neq i}^{N,N}$$

where $d_{S_iS_j}$ is the DTW distance between the speech sample i and j and represents the cost incurred to warp the i^{th} speech sample to j^{th} speech sample[2]. Note that there are a total of $\mathscr{N} = \left[\frac{N\times(N-1)}{2} - N\right]$ $d_{S_iS_j}$'s computed for N speech samples.

[1] depends on the number of frames in each speech sample

[2] We do not describe the DTW in this paper. It is also known as edit distance in computer science literature

3.1. Identifying the initial cluster centers and clusters

Step a The cluster center is identified by first determining $d_{S_kS_l}$, such that,

$$d_{S_kS_l} = \min_{i=0\cdots N, j=0\cdots N, j\neq i}\{d_{S_iS_j}\}$$

then the k^{th} speech sample S_k is chosen as the centroid (c_1).

Step b We then identify the first $\left(\frac{N}{n}-1\right)$ speech samples from the speech sample k by finding the $\left(\frac{N}{n}-1\right)$ minima in the set $d_{S_kS_i}$ for $i = 1,\cdots N$.

The first cluster is formed of the cluster centroid S_k and $\left(\frac{N}{n}-1\right)$ speech samples that are closest to S_k. Now the cluster C_1 has the cluster center S_k and other $\left(\frac{N}{n}-1\right)$ speech samples.

We repeat the process to identify all the n clusters, namely c_i and C_i for $i = 1,\cdots,n$. The cluster C_i for $i > 1$ is identified by eliminating all the speech samples in all the clusters $j < i$ and repeating Step (a) and Step (b).

3.2. Identifying the final clusters

For each speech sample which is not a cluster centroid (c_i for $i = 1,\cdots n$) we identify the true cluster centroid by

$$c_k = \min_{c_i}\{d_{c_i,S_j}\}$$

the speech sample S_j is assigned to the cluster C_k because it is closest to the speech sample c_k which is the cluster centroid.

3.3. Measuring Cluster Validity

There are three categories for testing the validity of clustering known as external, internal and relative criteria [6]. Given a data-set X and clusters C_k derived using a clustering algorithm on the data-set, then external criteria compares the obtained clusters C_k to a priori information on the clusters being known. For example, an external criterion can be used to examine the match between the cluster labels with the category labels based on a priori information. On the other hand the internal criteria evaluates the clusters based on the obtained clusters without any external information. Relative criteria compare the obtained cluster (C_k) with other clusters, obtained using different clustering algorithms.

In this paper we use only the external criterion to measure the validity of the clusters formed since we are aware of the expected clusters. Infact we can not use the (a) internal criterion because of the availability of only the relative distances between the samples and (b) relative criterion because to our best knowledge all the clustering schemes assume and work on absolute distances between the samples and not on relative samples. Specifically, we showcase the performance of the clustering algorithm in the form of a confusion matrix.

True Clusters → Formed clusters ↓	W_a	W_b	W_c
W'_a	1	0	0
W'_b	0.14	0.82	0.04
W'_c	0	0	1

Table 1. Confusion matrix for Wine data-set

True Clusters (→) Formed Clusters (↓)	I_s	I_{ve}	I_{vg}
I'_s	1	0	0
I'_{ve}	0	0.96	0.04
I'_{vg}	0	0.28	0.72

Table 2. Confusion matrix for Iris data-set

4. EXPERIMENTAL RESULTS

We first tested the performance of the proposed algorithm described in Section 3, on standard clustering data-sets [16] to establish the performance of the algorithm based on only the relative distances between the data-set samples. It is to be noted that the data-set that (unlike for speech samples) is available has (a) the same dimensional feature vector corresponding to the data-set and (b) absolute distance between the samples is available. But for purpose of evaluating the algorithm we considered only the relative distances between the objects in the data-set. The distances between the objects in the data set were computed using the euclidean distance rather than the DTW distance[3]. We used only the relative distance between samples to test the algorithm though we had access to the actual feature values.

We tested the performance the algorithm on 3 cluster test beds [16], namely (a) the wine cluster data set, which has 178 objects each represented by a feature vector of length 13 and in three classes/clusters; (b) iris data set which has 150 objects each represented by a vector of length 4 divided into 3 clusters and (c) Soya bean data set which consist of 47 objects each of dimension 35 divided into 4 classes. In each of the cases we start with the assumption that the number of clusters is known. Table 1 gives the cluster confusion for the 3 clusters. Note that W_a, W_b and W_c represent the true clusters while W'_a, W'_b and W'_c represent the clusters formed by our clustering scheme. It can be seen that both the clusters W_a and W_c have no misclustering (represented by 1) and form perfects clusters while there is a slight misclustering of objects belonging to cluster W_b. For the iris data-set the confusion in misclustering is shown in Table 2. It can be seen here that the formed clusters I'_s and I'_{ve} have been clustered very well; while a small portion of I'_{vg} have been clustered into I_{vg}. Table 3 shows the confusion matrix of clustering the Soya bean data-set into 4 clusters. It can be seen that the D'_1, D'_2 and D'_3 formed clusters have very good resemblance with the true clusters while D'_4 has some portions being misclustered as D_3.

[3]the objects had the same dimension

True Clusters (→) formed Clusters (↓)	D_1	D_2	D_3	D_4
D'_1	1	0	0	0
D_2	0	1	0	0
D'_3	0	0	0.8	0.2
D'_4	0	0	0.41	0.59

Table 3. Confusion matrix for small Soya bean data-set

True Clusters (→) Obtained Clusters (↓)	w_1	w_2	w_3	w_4	w_5
w'_1	0.8	0	0.1	0.1	0
w_2	0	0.9	0.05	0.05	0
w'_3	0	0	0.7	0.1	0.2
w'_4	0.05	0.05	0	0.9	0
w'_5	0.1	0.1	0	0	0.8

Table 4. Confusion matrix for speech signals

It should be noted that in all the experiments on the data-sets (Wine, Iris and Soya bean) were performed on the relative distances between the objects in the data-set and at no time was the absolute distance (though available) was used for the purpose of clustering. It is reasonable to expect a slight degradation in the performance of clustering because of the information used for clustering. The idea of conducting these experiments was to establish the performance of the clustering algorithm based on relative distance between objects/samples[4].

Table 4 shows the performance of the clustering algorithm on speech samples consisted of 20 utterance of 5 acoustically different words spoken by different people; a total of 100 speech samples. The idea is to establish if the speech samples can be grouped into 5 clusters corresponding to the 5 different words. In all a set of 14 features (8 MFCC, 4 Δ MFCC and 2 Δ^2 MFCC) were extracted for each frame. This representation of features was used to compute the DTW distance between a speech sample with every other speech sample ($d_{S_iS_j}$ in Section 3). Clustering was performed by first identifying the initial clusters, followed by identifying the final clusters. The performance is encouraging as seen in Table 4; but there have been reasonable misclustering especially with respect to w_3 cluster. We are in the process of experimenting with different feature sets to check if the clustering performance can be enhanced.

[4]The performance of our clustering scheme was similar to k-means clustering on the same data-set, but using absolute distances

5. CONCLUSIONS

In this paper, we have proposed a clustering technique which operates on relative distances between the objects and not on absolute distances. This scenario is prominent when clustering speech samples where feature vector representing the speech sample is not only of different length but there is no known absolute distances between all the speech samples. We showed the performance of the clustering technique on standard data-sets and established the methodology. We also showed the performance of the clustering technique to cluster real speech data. We are fine tuning the technique to come out with a mechanism to estimate the number of clusters so as to remove the assumption that the number of clusters be known apriori.

6. REFERENCES

[1] G. Forestier, C. Wemmert, and P. Gançarski, "Multi-source images analysis using collaborative clustering," *EURASIP J. Adv. Signal Process*, vol. 8, no. 2, pp. 1–11, 2008.

[2] P. Kumar, P. R. Krishna, R. S. Bapi, and S. K. De, "Rough clustering of sequential data," *Data Knowl. Eng.*, vol. 63, no. 2, pp. 183–199, 2007.

[3] S. M. Youssef, M. Rizk, and M. El-Sherif, "Enhanced swarm-like agents for dynamically adaptive data clustering," in *CEA'08: Proceedings of the 2nd WSEAS International Conference on Computer Engineering and Applications*. Stevens Point, Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS), 2008, pp. 213–219.

[4] A. Azran and Z. Ghahramani, "A new approach to data driven clustering," in *ICML '06: Proceedings of the 23rd international conference on Machine learning*. New York, NY, USA: ACM, 2006, pp. 57–64.

[5] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Comput. Surv.*, vol. 31, no. 3, pp. 264–323, 1999.

[6] R. Xu and D. Wunsch, *Clustering*. Wiley-IEEE Press, 2009.

[7] G. Vensko, K. B. Lieu, S. A. Meloche, and J. C. Potter, "Dynamic time warping (dtw) apparatus for use in speech recognition systems," *US Patent 5073939*, December 17, 1991.

[8] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Transactions on Communications*, vol. 28, no. 1, pp. 84âĂŞ–95, 1980.

[9] A. Buzo, A. H. Gray, R. M. Gray, and J. D. Markel, "Speech coding based on vector quantization," *IEEE Transactions on Accoustics, Speech and Signal Processing*, vol. 28, no. 5, pp. 562âĂŞ–574, 1980.

[10] A. Banerjee, S. Merugu, I. Dhillon, and J. Ghosh, "Clustering with Bregman divergences," *Proceeding of SIAM Data Mining conference*, pp. 234–245, 2004. [Online]. Available: citeseer.ist.psu.edu/article/banerjee04clustering.html

[11] H. Jin, F. Kubala, and R. Schwartz, "Automatic speaker clustering," in *DARPA Speech Recognition Workshop*, 1997, pp. 108–111.

[12] S. Molau, M. Pitz, R. S. Uter, and H. Ney, "Computing mel-frequency cepstral coefficients on the power spectrum," *Proc. Int. Conf. on Acoustic, Speech and Signal Processing*, pp. 73 – 76, 2001.

[13] T. F. Quatieri, "Discrete-time speech signal processing: Principles and practice," *Pearson Education*, vol. II, pp. 686, 713, 1989.

[14] CMU, "http:// cmusphinx.sourceforge.net/ sphinx4/ javadoc/ edu/ cmu/ sphinx/ frontend/ frequencywarp/ melfrequencyfilterbank.html."

[15] S. Sigurdsson, K. B. Petersen, and T. L. Schiøler, "Mel frequency cepstral coefficients: An evaluation of robustness of mp3 encoded music," *Conference Proceedings of the Seventh International Conference on Music Information Retrieval (ISMIR)*, Vicoria, Canada, 2006.

[16] A. Asuncion and D. Newman, "UCI machine learning repository," 2007. [Online]. Available: http://www.ics.uci.edu/∼mlearn/MLRepository.html