Speaker and Language Recognition by GMM

Darshan Ramakant Bhat (MT2015038)
Freeze Francis (MT2015044)
Mohammed Haroon D (MT20150650)
IIIT Bangalore

Abstract

A good Speaker Recognition (SR) system requires a robust feature extraction unit followed by a speaker modeling scheme for generalized representation of these features. Over the years, Mel-Frequency Cepstral Coefficients (MFCC) modeled on the human auditory system has been used as a standard acoustic feature set for SR applications. We build a robust Speaker Recognition system by extracting MFCC features from the voice samples and fitting a Gaussian Mixture Model for each speaker. MFCC features are extracted from the new test sample and likelihood is calculated for each model to identify the speaker. The speaker repeating most in the voice frames is predicted as the speaker of the test voice sample. Similar approach is followed for language recognition system.

1 Problem Statement

Voice samples of 22 different speakers is given. We need to fit a Gaussian Mixture Model for each speaker and be able to predict the speaker in a new test sample using this model. Similar dataset of voice samples in different languages is given. We need to build a similar system to be able to recognise the language from a new test sample.

2 Dataset

2.1 Speaker Recognition

Training

Number of Speakers : 22
Voice samples per speaker : 8

Test

Number of Speakers : 22
Voice samples per speaker : 2

2.2 Language Recognition

Training

Number of languages : 4
Languages : Bengali, Kannada, Odia, Telugu
Voice samples per language : 16

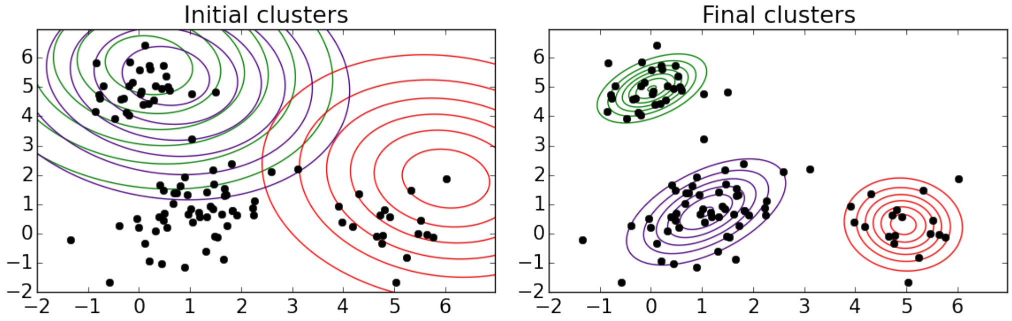
Test

Number of languages : 4
Languages : Bengali, Kannada, Odia, Telugu
Voice samples per language : 4

3 MFCC coefficients

In sound processing, the mel-frequency cepstrum (MFC) is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency. MFCCs are commonly used as features in speech recognition[6] systems, such as the systems which can automatically recognize numbers spoken into a telephone. MFCC takes human perception sensitivity with respect to frequencies into consideration, and therefore are best for speech/speaker recognition.

4 Gaussian Mixture Model



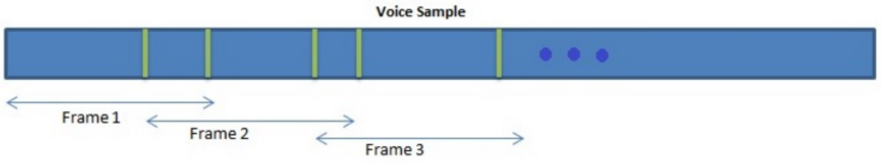
A Gaussian mixture model is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with unknown parameters. One can think of mixture models as generalizing k-means clustering to incorporate information about the covariance structure of the data as well as the centers of the latent Gaussians. Gaussian mixture models are used a lot when the underlying populations can be explained by a normal distribution and there are many heterogeneous populations. The main difficulty in learning Gaussian mixture models from unlabeled data is that it is one usually doesnt know which points came from which latent component (if one has access to this information it gets very easy to fit a separate Gaussian distribution to each set of points). Expectation-maximization is a well-founded statistical algorithm to get around this problem by an iterative process. First one assumes random components (randomly centered on data points, learned from k-means, or even just normally distributed around the origin) and computes for each point a probability of being generated by each component of the model. Then, one tweaks the parameters to maximize the likelihood of the data given those assignments. Repeating this process is guaranteed to always converge to a local optimum.

5 Procedure

5.1 Speaker Recognition

Training:

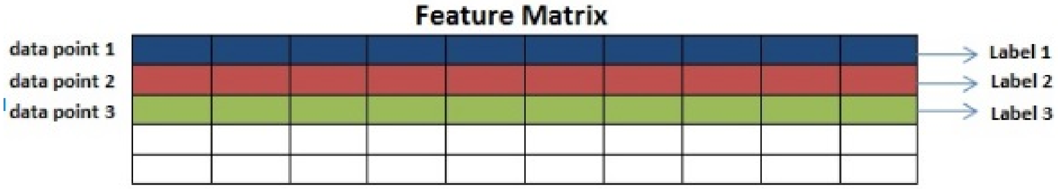
1. Extraction of MFCC features from the voice sample
    (a) We divide the each voice sample into overlapping frames of 25ms each.



    (b) 42 MFCC coefficients are extracted from each frame.
    (c) This forms a feature matrix for a given voice sample. Each row in the feature matrix corresponds to a datapoint (corresponding to a frame) in a 42 dimensional feature space.
    (d) Similar feature matrix is formed from the MFCC coefficients for other voice samples of the same speaker. These features are stacked one below the other to form one big matrix for a specific speaker. This has the same effect as concatenating all the voice samples as one big voice sample for a specific speaker and extracting MFCC features by dividing it as frames
    (e) Similar procedure is followed for each speaker to get a big feature matrix for each speaker.

2. A gaussian mixture model is fit on each matrix. This model will be in 42 dimensional space. This way each speaker is described by a unique Gaussian Mixture Model which can be used to identify the speaker. The number of gaussians to fit is a hyperparameter to be set manually. We tried various number of gaussians to determine the optimal number.

Inference



1. MFCC features are extracted as in the training phase to form a big matrix for the given test voice sample.

2. Each row is a datapoint in the feature space. A single row is taken and its likelihood with respect to all the 22 different speakers is calculated using their GMM model.

3. The speaker with the highest likelihood is assigned as a label for the datapoint. This is continued for the remaining rows in the feature matrix of test voice sample.

4. Now we have a vector of labels for the matrix.

5. The particular label repeating most in the vector is predicted as the speaker of the test voice sample.

6. This is done for remaining test voice samples and accuracy is calculated.

5.2 Language Recognition

Training

1. MFCC features are extracted from the voice samples by dividing into frames of 25ms each.

2. We get 42 dimensional feature matrix for each language.

3. A gaussian mixture model is fit on each matrix. This way each language is described by a unique Gaussian Mixture Model which can be used to identify the language of the speaker. The number of gaussians to fit was tried with various number to determine the optimal number.

Inference

1. MFCC features are extracted as in the training phase to form a matrix in 42 dimensions for a voice sample.

2. A single row is taken and its likelihood with respect to all the 4 different languages is calculated using their GMM model.

3. The language with the highest likelihood is assigned as a label for each row in the matrix to get a vector of labels for the matrix.

4. The particular label repeating most in the vector is predicted as the language of the test voice sample. This is done for remaining test voice samples and accuracy is calculated.

6 Implementation

Matlab is used for this assignment. ”Voicebox” toolbox is used to extract MFCC features and fit the GMM. **train_GMM.m** is the file containing code to extract MFCC features and fit GMM for each speaker. The number of gaussians are experimentally varied and 32 Gaussians for each model is found to be the optimal parameter. **speaker_recognition_main.m** uses the trained model to predict the test voice sample and calculates the training and testing accuracy.

Similarly **language_recognition_main.m** is used for language recognition. While running the code, make sure that folder paths are given appropriately. As of now, it is hardcoded.

7 Results

1. **Speaker Recognition:**
   Training Accuracy : 100%
   Test Accuracy : 95.45% was the maximum on 5 runs.

2. **Language Recognition:**
   Training Accuracy : 100%
   Test Accuracy : 93.78% was the maximum on 5 runs.

8 Conclusion

We found that MFCC coefficients can be used as good representational features for voice data. Accuracy of the system is sensitive to the number of MFCC features. Hence this hyperparameter should tuned appropriately after experimenting with various numbers. These features along with GMM for each speaker can be effectively used for Speaker identification as well as Language identification tasks. We noticed that another hyperparameter here is the number of gaussians to fit in the GMM which needs to be tuned properly. With proper tuning, GMM along with MFCC features can be used to model the voice data with reasonable accuracy.

8.1 References

[1] Reynolds, Douglas A., and Richard C. Rose. ”Robust text-independent speaker identification using Gaussian mixture speaker models.” IEEE transactions on speech and audio processing 3.1 (1995): 72-83.

[2] Reynolds, Douglas A., Thomas F. Quatieri, and Robert B. Dunn. ”Speaker verification using adapted Gaussian mixture models.” Digital signal processing 10.1 (2000): 19-41.