Voice Controlled Ordering and Billing System of BUET Cafeteria

Tusher(1806174) Swmic(1806175) Nafiul(1806176) Rudmila(1806190)

4 September 2022

Abstract

In this project we have build voice Controlled Ordering and Billing System of BUET Cafeteria.The system will take two speech signals from a user, one for the name of the item and one for the quantity, and compare them to previously recorded inputs from different users.When the system finds a suitable match, it determines which item the person wishes to purchase as well as the quantity. The idea is to start by extracting useful features from raw speech data collected from various users. The Mel-Frequency Cepstral Coefficients algorithm was used in this paper to extract these features, which involves calculating coefficients from user audio data that are unique to each user. Then, using the K-NN Algorithm, these coefficients are compared to new MFCC coefficients extracted from audio data from a new user. K-nearest neighbors (KNN) algorithm uses 'feature similarity' to predict the values of new data points which further means that the new data point will be assigned a value based on how closely it matches the points in the training set. This paper's precision is moderate.

Contents

List of Figures

Chapter 1

Introduction

In BUET Cafeteria, there is a shortage of manpower to collect order from mass student. Students had to wait for longer periods to place their order. In this project, we want to automate the process of placing orders. We have used algorithms like MFCC and dtw to detect voice message and find its equivalent word.

Converting Speech recognition has become an increasingly popular concept over the years because it provides an option to detect human voice and convert it into machine-readable format for further use. Ability to distinguish one person's voice from other allow us to compare and reach a conclusion.

Chapter 2

Methodology

Workflow of the entire process can be divided in several segments.

a) Pre-Processing Data, b) Taking voice input and perform feature extraction, c) comparing features with previously stored data features.

2.1 Pre-processing Data

An end point detection algorithm is used to separate the main energy content of the signal from the rest. It sets a threshold below which it considers audio amplitude as non-voice or unvoiced data. Then it goes over the whole signal and tries to identify which regions in the raw audio data can be considered useless. It basically gives a noise removed, or silence removed signal from the raw audio data.

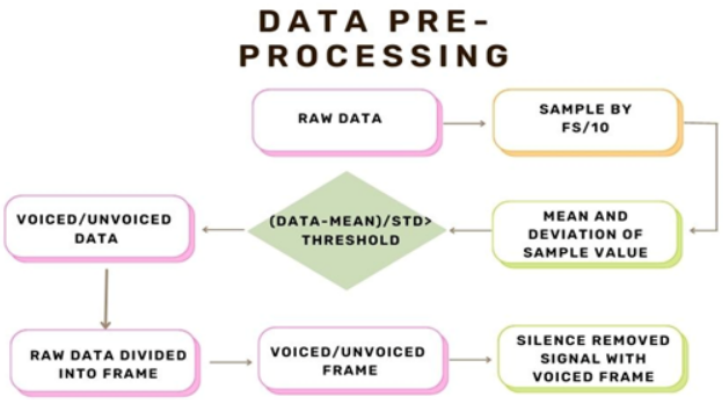

Figure 2.1: Pre-processing Data

2.2 Feature extraction

It is the core of entire project. Some discrete values representing the silence removed signals, or the energy to be exact need to be generated and then classifiers can be applied to them. There are many feature extraction techniques like Linear Predictive Coding (LPC), Perceptual Linear Prediction (PLP), Relative Spectral (RASTA), Discrete Wavelet Trans- form (DWT), Wavelet Packet Transform (WPT), Probabilistic Linear Discriminate Analysis (PLDA), and Mel-Frequency Cepstral Coefficient (MFCC). Here we have used MFCC algorithm in this project.

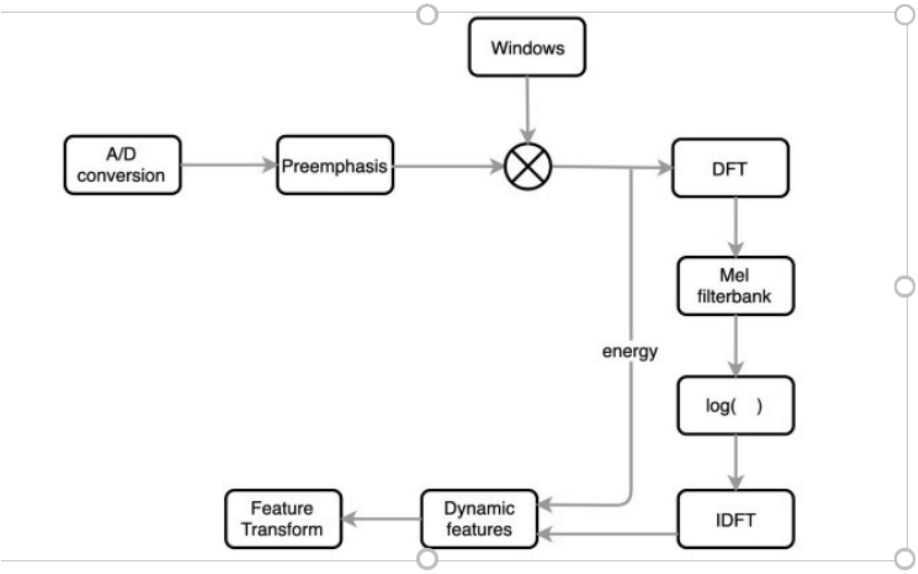

Figure 2.2: Feature extraction

2.2.1 Dividing data into small time frames

Audio is divided into parts for better classification of power spectrum of signal. Suppose we will not take 'Burger' rather we will rake 'Ba','r','g','ar' as voice segment to compute power of individual section. Here the frame width is taken to be 25ms and shifting 10ms.

2.2.2 Calculation of power spectrum of each frame

After signal is divided into frame, we pass signal to hamming window. Audio input may have different duration but here we are calculating 2048 point DFT and thus power spectrum is derived from DFT points.

Sframe(k)=N1k=0sframe(n)×hamming(n)×ej2knN Sframe(k)=k=0N1sframe(n)×hamming(n)×ej2knN

Power spectrum of frame, Pi(k)=Nk=0—Si(k)—2 Pi(k)=k=0N—Sik—2

2.2.3 Applying MEL-filter bank on power spectrum

Mel-frequency is another frequency scale which is used to relate the perceived frequency to the measured frequency of audio signals. Mel-frequency makes frequency difference relationships linear, close to how humans hear differentiate these frequencies. In MEL- domain frequency is related to Hz,

f=700×(eM(f)11251) f=700×(eM(f)11251)

.

From 2048 DFT points, we apply triangular filter bank on 1025 point. For 26 frame, we have applied 26 triangular filter bank to power spectrum.

2.2.4 Taking log of energy

After the filter banks are applied, then the log of the filtered spectrum of each frame is taken. The amount of energy needed to be put into an audio signal to make it sound double as loud as before is not double that of the previous energy. The relation between loudness and frequency components present is not linear. The loudness of 300Hz -3400Hz contents do not increase in a linear manner, rather in an exponential manner. So, we take log of the power spectrum.

2.2.5 Discrete Cosine Transform (DCT):

The discrete cosine transform of the log of each power spectrum corresponding to the 26 frames that we are working with gives us 13 coefficients. The DCT matrix is a 13×26 matrix. This matrix helps to decorrelate the power spectrum overlap between the filter banks. Now, these are the MFCC coefficients to which classifier will be applied.

2.2.6 Delta and Delta-deltas:

The mfcc coefficients only contain information of the power spectrum of a single frame. But there might also be information of speech inherent in the dynamics or trajectories of mfcc over time. For that we calculate delta and delta-delta coefficients

dt=Nn=1n(ct+nctn) 2Nn=1n2 dt=n=1Nn(ct+nctn) 2n=1Nn2

dt is the delta coefficient of frame t computed from static coefficients ct+N to ct-N. N is usually chosen as 2. The mean of all the values of a particular MFCC coefficient for all frames is taken before calculating the delta and the delta-delta coefficients.

2.3 Classifier

After feature extraction, we use classifier to measure similarity and dissimilarity to detect which order was placed. For classification, various techniques are available like Gaussian Mixture Model (GMM), Hidden Markov Model (HMM), Artificial Neural Network (ANN), Dynamic Time Warping (DTW), Vector Quan-

tization (VQ) models, Support Vector Machine (SVM), and k Nearest Neighbor (kNN). In this project, we have used KNN (K-Nearest Neighbour) algorithm to measure similarity.

2.3.1 K-NN Algorithm:

K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories. K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.

The K-NN working can be explained on the basis of the below algorithm:

Step-1: Select the number K of the neighbors Step-2: Calculate the Euclidean distance of K number of neighbors Step-3: Take the K nearest neighbors as per the calculated Euclidean distance. Step-4: Among these k neighbors, count the number of the data points in each category. Step-5: Assign the new data points to that category for which the number of the neighbor is maximum. Step-6: Our model is ready.

Here, we have used four cluster for burger, pizza ,tea ad coke.

Chapter 3

Result and Discussion

New user data is collected and its MFCC coefficients are calculated after storing the audio train files in a new directory. The MFCC coefficients for train audio data are then computed. The newly calculated coefficients are then compared to the previously calculated coefficients for a single user to determine their differences or degree of dissimilarity. The best match is determined by taking the minimum of all these comparisons.In other words, the new audio data most closely matches the previously saved data. This procedure is followed for both item and quantity selection. If the new audio data matches the existing data in both cases, we can confirm that it is the item and quantity that the person ordered. The bill is then generated.

3.1 percentage of Accuracy

The folloing graphs show the percentage of successful and failed detection train By 40 Samples.

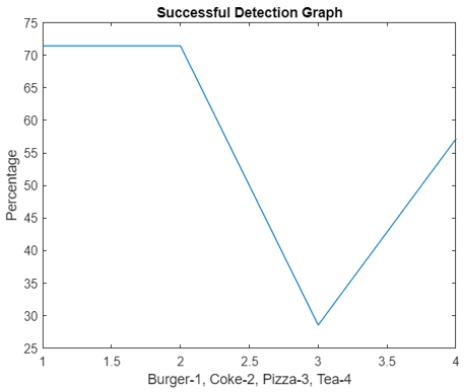

Figure 3.1: Successful Detection

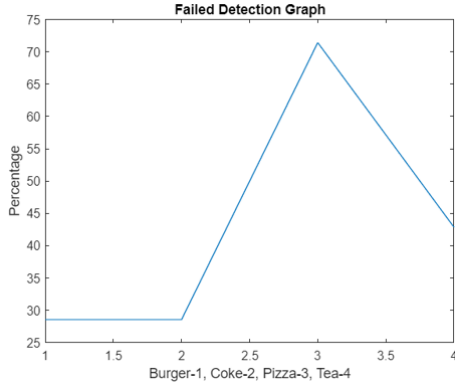

Figure 3.2: Filure Detection

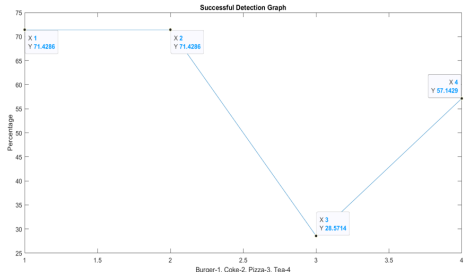

Figure 3.3: Successful percentage

Percentage: 71.4286 71.4286 28.5714 57.1429 Here we get higher success rate for 1(Burger),2(Coke) and lower success rate for Pizza(3). We get average detection for Tea.

3.2 Noise Performance-Order Name

Here, we add noise to the audios and here is the plot of the successful detection of the order's names.

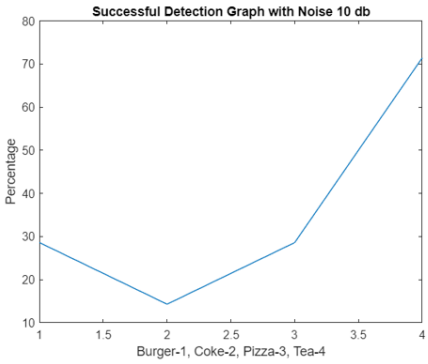

Figure 3.4: with noise 10dB

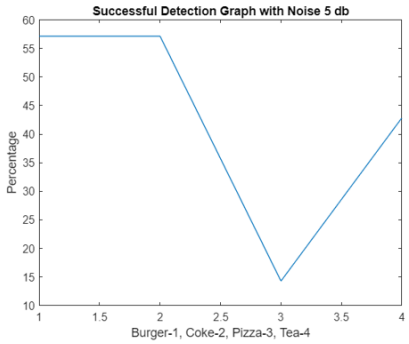

Figure 3.5: with noise 5dB

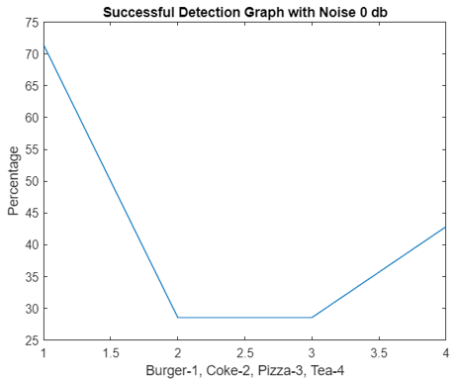

Figure 3.6: with noise 0dB

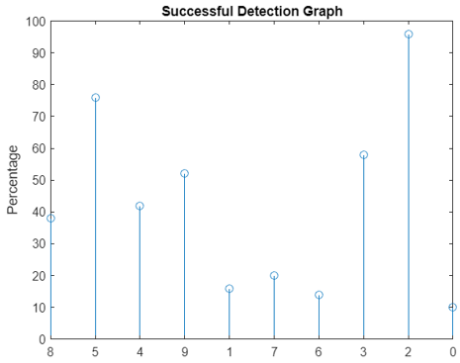

Figure 3.7: Successful percentage

Percentage = 1×10
96 20 16 10 76 58 14 52 42 38

3.3 Noise Performance-Numbers

Here, we add noise to the audios and here is the plot of the successful detection of the order's quantity.

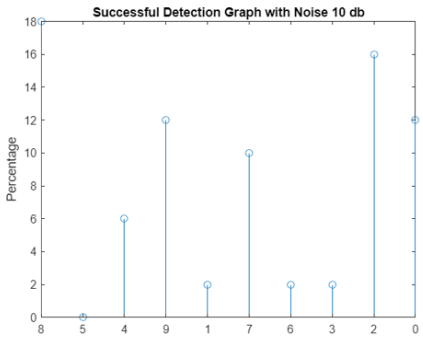

Figure 3.8: with noise 10dB

Percentage = 1×10
32 18 6 14 6 0 10 6 4 2

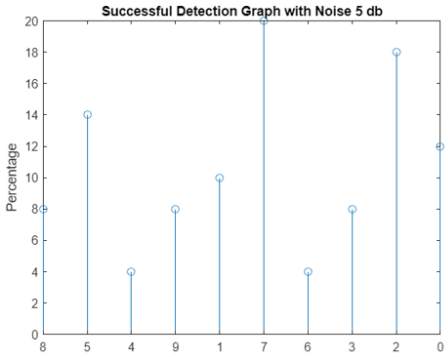

Figure 3.9: with noise 5dB

Percentage = 1×10
18 20 10 12 14 8 4 8 4 8

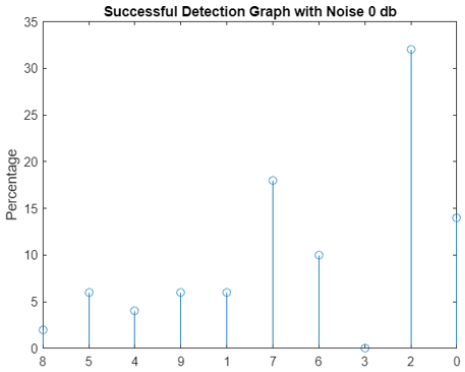

Figure 3.10: with noise 0dB

Percentage = 1×10
16 10 2 12 0 2 2 12 6 18

3.4 Confusion Matrix

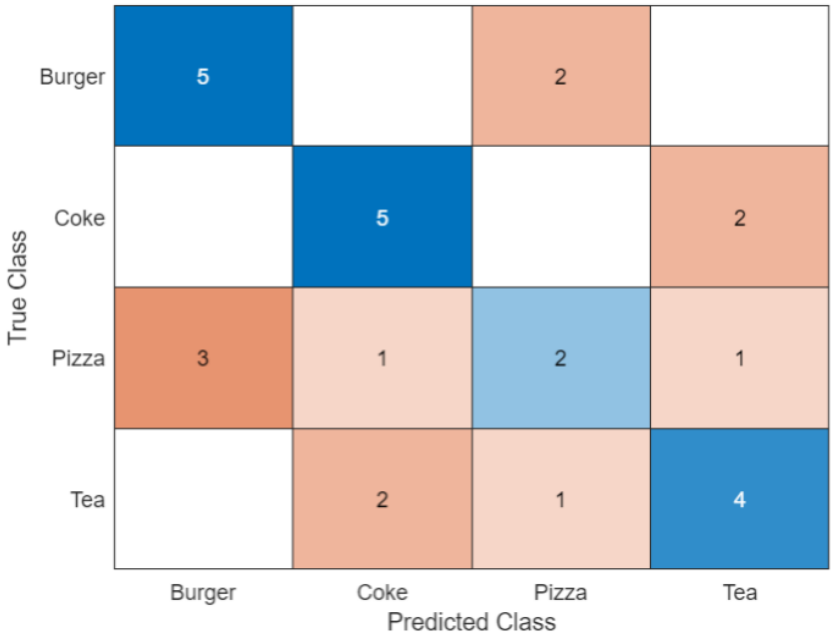

Figure 3.11: For order's name

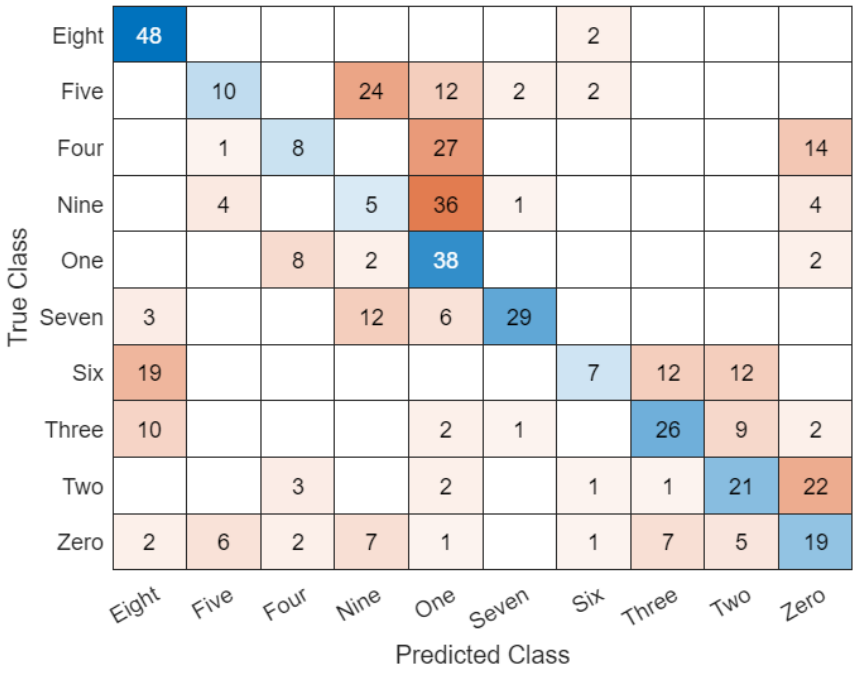

Figure 3.12: For order's quantity

Chapter 4

Conclusion

The idea of this project was voice Controlled Ordering and Billing System of BUET Cafeteria. After applying feature extraction technique using MFCC, test audio data was applied for verification. The verification part was done by proper handling of KNN algorithm.

4.1 Limitations

The accuracy level was moderate. Accuracy is not 100 percent Due to-

1.Shortage of data.
2.Speech spectrum may overlap.
3.Noise can't be entirely removed.
4.Device Input error.
5.It shows less accuracy in noisy environment.

4.2 Future Scope

Hence, we had to concede with the results obtained. But, the results were still modest and leaves room for future improvement if needed.

1. We can take more data from the people to increase accuracy.
2. More sophisticated Noise reduction process could be implemented.
3.Using more sophisticated classifier algorithms such as Gaussian Mixture Model (GMM)

Chapter 5

References

[1] Wei Han, Cheong-Fat Chan, Chiu-Sing Choy and Kong-Pang Pun, "An efficient MFCC extraction method in speech recognition," 2006 IEEE International Symposium on Circuits and Systems (ISCAS), 2006, pp. 4 pp.-, doi: 10.1109/ISCAS.2006.1692543.

[2] Chakraborty, K., Talele, A., Upadhya, S. (2014). Voice recognition using MFCC algorithm. International Journal of Innovative Research in Advanced Engineering (IJIRAE), 1(10), 2349-2163.