INTRODUCTION TO SPEECH AND LANGUAGE PROCESSING TERM PROJECT

EARWIG

SEFA ALP

ATAKAN PEHLİVANOĞLU

ABSTRACT

We make important meetings with our boss, teachers, or partners occasionally. These meetings can happen for solving certain problems

or drawing a new roadmap for the project that we are working on. But important details of these meetings are forgotten in several days and

the working process can be staggered due to missing information.

In this project, we focused on this problem and trying to find out an easier and effective method than taking notes. Earwig occurred in

this manner. In its simplest form Earwig first takes speaker's voice records separately for creating training. Using this voice records program to train itself for each speaker.

Then we record dialogue or meeting to the program. And output program gives the sentences as voice record that each person in the speech used according to their speaking order.

INTRODUCTION

In this Project, we are going to see building blocks of our speaker recognition program this way we can analyze program with looking its smaller part. Also, discuss which method we use and try to understand their working mechanisms. Methods will be compared in this paper, to understand their efficiency and effectiveness for our case
If we divide our program into fundamental steps :

1. Record participants speech signal via GUI for training data
2. Splitting all speech into frames
3. MFCC voice extraction
4. Training data via FNN method
5. Recording the real dialogue after training each participants speech
6. Extracting all sentences from recording using Auto Correlation and Short-TEnergy
7. Splitting sentences into frames and extracting their MFCC features.
8. Predict which sentences belong to which people via trained FNN classification.

In the Project definition, the whole process will be explained by highlighting the important points to make the explanation simple and efficient. Technical detail can be found in theoretical background and testing parts.

PROJECT DEFINITION

Here we are going to see the program's working steps and how speech recognition techniques used in these steps.

The program differs participant's voices according to their recorded speeches before the meeting. When the program starts the program asks for the first participant to speak for him/her voice analysis. Each participant speaks for more than thirty seconds. The more the participant speaks, the more data we have, and training data gets expanded thus program could train better, and accuracy increases. After obtaining speech records we need to split data into small frames for data analysis. After splitting data into frames a matrix created that includes small speech frames for each speech.

Let's say our speech records occurred from 360 small frames that include 480 samples each of them, so our matrix's size would be [360][480].

Then we remove silence parts in these speeches to get pure fragments that include only speech in it.

The actions we have done so far, have been to obtain the frames we will use to extract the speaker's voice characteristic. So, now we have

speech frames that include only voice in it. Right now one of the most important steps of the algorithm is going to be used, MFCC. MFCC is a method

that extracts a person's vocal tract shape according to the speech signal. Why we need vocal tract information? When we speak vocal tract shape changes according

to covering muscles use that surrounding vocal tract. Also, genetically each human has its vocal tract shape and that's the other reason why we have unique voices.

So, vocal tract shape gives us specific information for each person and with using MFCC we are going to extract information. For each frame for each person, we extract

MFCC features and collects them in the array. Than we concatenated both MFCC feature arrays thus we obtain training input data. Also according to speaker id, we created

training output. Let say two people speaking in our case so the first person would be labeled as "0" and the second person is "1" so each MFCC features get its output value as its label.

So far, we have seen how the sound is divided into small pieces for each speaker and the MFCC features of each frame are obtained for creating a training dataset.

In this way, we obtained a training dataset for each participant. So we can start the training process. Before the training process, we need to talk about the machine learning technique

that we use briefly. We used feed-forward neural network (FNN) in this project. FNN is a neural network model inspired by the human brain that has nodes and node layers like neurons and synapses in the brain.

We used training data which includes MFCC features for each person's speech frames as input and trained FNN.

Now, we are going to see how meeting dialog can be divided into sentences according to their labels. First, we split the dialog record into frames. Then, to obtain sentences in the record, voiced and unvoiced frames should be detected. In this part, we are going to use

STE and Autocorrelation functions. STE refers to short-time energy, it simply takes the square of each sample and sums them up for each frame. If the frame has low STE energy

that means most probably there is no voice in this frame if STE value is high vice versa. STE is the first voice detection algorithm we use autocorrelation function (STAC) too.

Autocorrelation represents the relationship of a signal for different time values in the frame. So, It is used to detect repeating patterns in sound analysis. If the signal is periodic

autocorrelation functions first coefficients are getting a higher value. Voiced frames have periodic signals in it thus their autocorrelation coefficients are higher.

According to both STAC and STE values voiced and unvoiced parts have been found. To split speech into sentences we wrote an algorithm that checks if there are more than 10 unvoiced frames

between two voiced frames these areas are splitting. So, we find a sentence ending and beginning according to unvoiced frames count.

Right now, we obtained sentences in the speech recording and collected them in the array. Now we need to find their labels using artificial intelligence.

Here we are going to split each sentence in frames and find their MFCC features and collected them in the array. This array is going to represent our test dataset. Via using test dataset we are going to predict each

sentence label. For example, if a sentence occurs from 6 frames and FNN predicts 5 of them as "1" and 1 of them as "0". The sentence would be labeled as 1. that is, the sentences are tagged according to the weight of the labels of the frames it contains.

THEORETICAL BACKGROUND

While carrying out this project, we tried many different techniques and used the best results among them. In this section, we have included detailed explanations for a better understanding of all the techniques we use.

We used a total of 4 methods to separate the voiced and sun-voiced parts of the speech, but the most effective of these were STE and STAC.

SHORT TIME ENERGY

This method used for finding energy of short speech segment. Calculated as below equation.

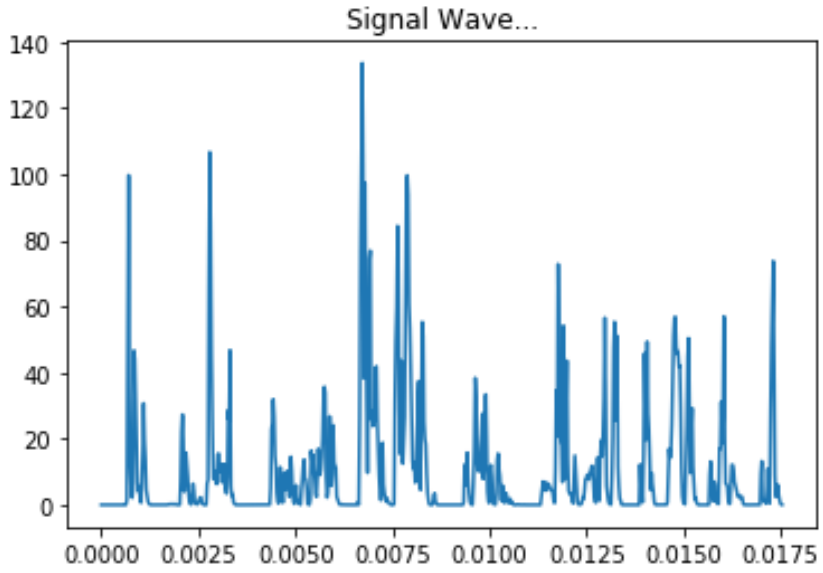
$$E = \sum_{m=-\infty}^{\infty} x^2[m]$$

this is the long term definition of signal energy

there is little or no utility of this definition for time-varying signals

$$E_{\hat{n}} = \sum_{m=\hat{n}-L+1}^{\hat{n}} x^2[m] = x^2[\hat{n}-L+1] + \ldots + x^2[\hat{n}]$$

Figüre 1: Calculation formula of STE [1]



Figüre 2: STE values according to frames in speech signal

- If signal includes speech in it that means it also has higher energy in that region.
- So, if we find energy of each frame, we obtain short-term energy array.
- Thus, we can differ speech and non-speech frames.

SHORT TIME AUTOCORRELATION

Autocorrelation, also known as **serial correlation**, is the correlation of a signal with a delayed copy of itself as a function of delay. Informally, it is the similarity between observations as a function of the time lag between them.[2]

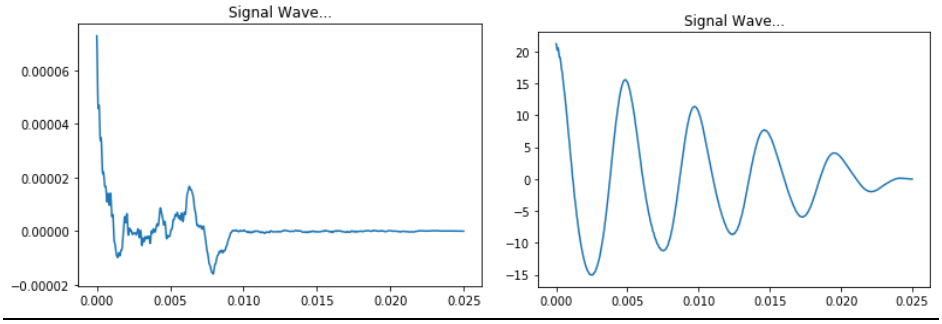


Figure 3: Autocorrelation coefficiens of unvoiced (left) and voiced (right) signal.

So, It is used to detect repeating patterns in sound analysis, that is, determines whether the signal is periodic or not. Periodic signals include voice and non-periodic signals don't, this way we can differ voiced and unvoiced frames.

After detecting voiced frames, MFCC used for feature extraction from these frames.

MEL-FILTER-CEPSTRAL-COEFFICIENTS (MFCC)

- Refers to mel frequency cepstrum coefficients.
- This coefficients represent phonemes as the shape of the vocal tract.

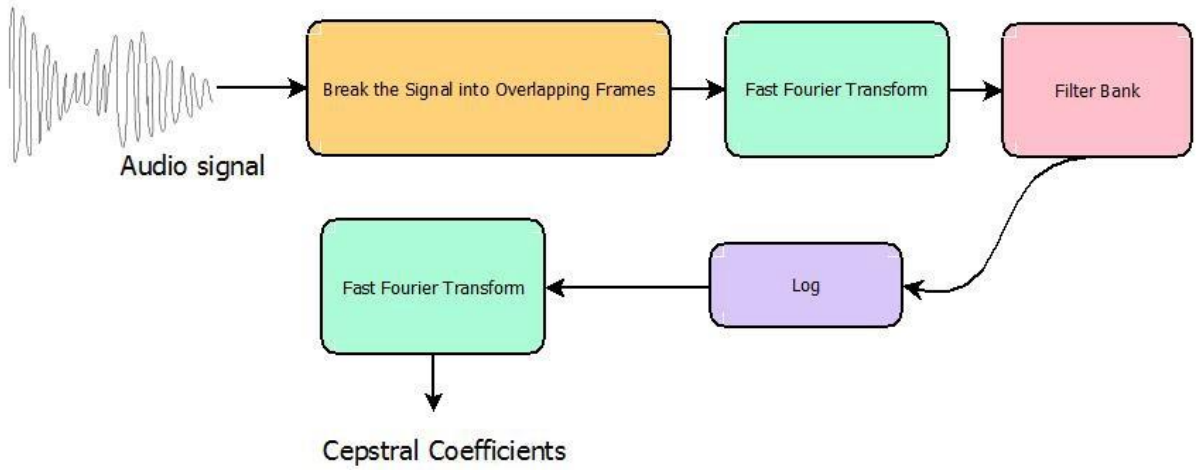


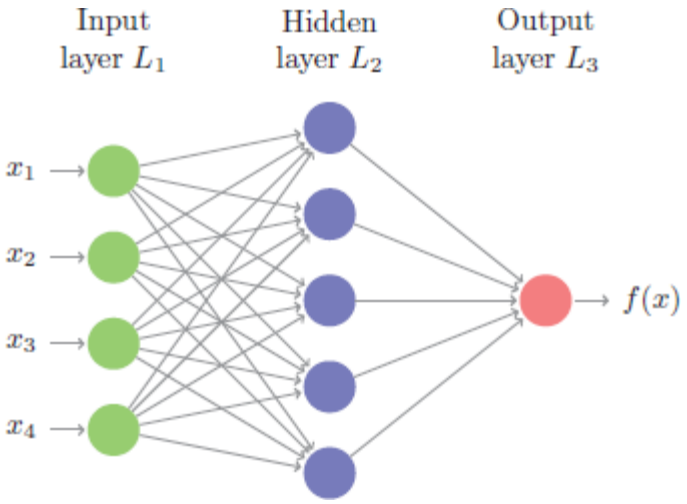
Figure 4: Shows how to obtain mel-frequency cepstrum coefficient.[3]

1. Frame the signal into short frames.
2. For each frame calculate the periodogram estimate of the power spectrum.
3. Apply the mel filterbank to the power spectra, sum the energy in each filter.
4. Take the logarithm of all filterbank energies.
5. Take the DCT of the log filterbank energies.
6. Keep DCT coefficients 2-13, discard the rest.

[4]

FEED FORWARD NEURAL NETWORK (FNN)

The feedforward neural network was the first and simplest type of artificial neural network devised. In this network, the information moves in only one direction, forward, from the input nodes, through the hidden nodes (if any) and to the output nodes. There are no cycles or loops in the network. [5]



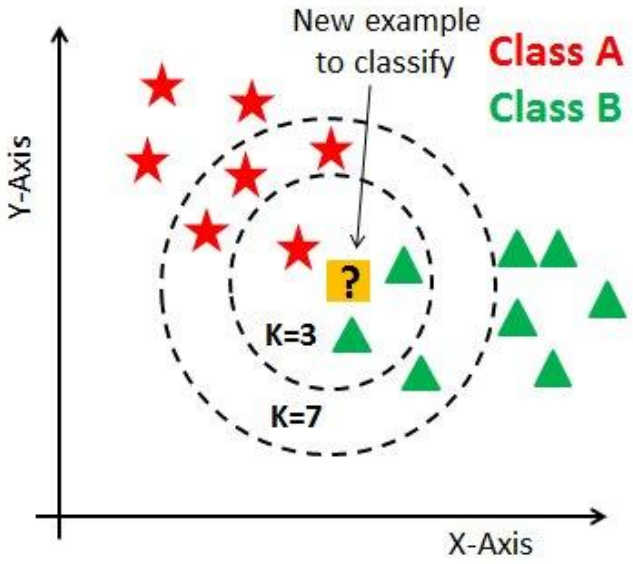
Figüre 5: FNN model basically represented. [6]

K NEAREST NEIGHBOR (KNN)

A case is classified by a majority vote of its neighbors, with the case being assigned to the class most common amongst its K nearest neighbors measured by a distance function. If K = 1, then the case is simply assigned to the class of its nearest neighbor. [7]

Euclidean $\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$

Figüre 6: Euclidean distance method used generally for distance calculation.[8]



Figüre 7: Euclidean distance method used generally for distance calculation.[9]

IMPLEMENTATION AND TESTING

We used two different artificial intelligent approaches to make predictions on different cases. First approach was KNN method because according to the FNN, KNN is much simpler model for implementation and testing.

Actually, according to its basic algorithm, the results were very effective. We tested the program on between man-man and man-woman to understand its accuracy between genders. In this case, accuracy represents correctly predicted frame count in the speech signal. The program divides speech signal into 25 ms frames and each frame predicted according to its MFCC coefficients. Let's say our speech record occurs from 40 frames and the program predicted 36 frames truly. In this case, accuracy is %90. Let's see the real results for method comparison.

K NEAREST NEIGHBOR (KNN) RESULTS

We did all the tests among same three people to be fair.

Woman – Man voice splitting :

Accuracy : %86

Man – Man voice splitting :

Accuracy : %90

We did all the tests among the same three people to be fair.

With KNN program could predict sentences label so successfully as %86 for woman to man conversation and %90 for man to man conversation. Since we coincide with the time of the corona pandemic, we had to conduct tests with very few people, but it is extremely promising and satisfying to make such accurate decisions in the dialogues between the same family members, that is, the mother-son and father-son in the example here.

After these successful results, we wanted to observe the results of the program with more complex and strong algorithms such as feed-forward neural networks.

FEED FORWARD NEURAL NETWORK (FNN) RESULTS

The same three-person used to make tests fair and understand algorithms of real capability.

Woman – Man voice splitting :

Accuracy : %100 Loss : 0.0016

Man – Man voice splitting :

Accuracy : %97 Loss : 0.0548

FNN algorithm worked nearly perfectly! Our program predicted speech between woman and man correct so accuracy was %100 and the loss function give very small value such 0.001. Also, the conversation between man and man correctly separated with again nearly perfect accuracy such %97 also loss function was a little bit higher than woman – man comparison but still close to 0 as 0.05. The algorithm took 40 speech frames and predicted truly all for the woman-man case and missed only 1 instance for the man-man case.

CONCLUSION

In this project, we have designed a program that divides the voice recording into sentences, then labels them according to the participants in business or college meetings. The program takes the participant's voice records and splits them into short frames. Using voice- unvoice detection algorithms such STE and STAC program only used voiced parts for feature extraction. Then, extract the speaker's voice features using MFCC algorithm. Then, using these MFCC features for each participant create one-bit training input data and define training output data as their label numerically (0-1 …). Using this training data train the feed-forward neural network model. After model training, the program splits conversation record into sentences and predict labels of these sentences according to the voice features of their short frames with using a feed-forward neural network model. if the sentence frames are tagged predominantly 1, the sentence is tagged as 1, if it has predominantly 0, the sentence is tagged as 0. This way program labeled sentences according to its speaker's id.

Thus, we obtained a useful program that ensures that no details are skipped during important meetings and that the participants do not waste time taking notes.

REFERENCES

[1]

Ece.ucsb.edu. 2020. [online] Available at: <https://www.ece.ucsb.edu/Faculty/Rabiner/ece259/digital%20speech%20processing%20course/lectures_new/Lectures%207-8_winter_2012.pdf> [Accessed 31 May 2020].

[2]

En.wikipedia.org. 2020. *Autocorrelation*. [online] Available at: <https://en.wikipedia.org/wiki/Autocorrelation> [Accessed 31 May 2020].

[3]

Medium. 2020. *The Dummy'S Guide To MFCC*. [online] Available at: <https://medium.com/prathena/the-dummys-guide-to-mfcc-aceab2450fd> [Accessed 31 May 2020].

[4]

Practicalcryptography.com. 2020. *Practical Cryptography*. [online] Available at: <http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/> [Accessed 31 May 2020].

[5]

En.wikipedia.org. 2020. *Feedforward Neural Network*. [online] Available at: <https://en.wikipedia.org/wiki/Feedforward_neural_network> [Accessed 31 May 2020].

[6]

Uc-r.github.io. 2020. [online] Available at: <https://uc-r.github.io/public/images/analytics/deep_learning/fig18_1.png> [Accessed 31 May 2020].

[7]

Saedsayad.com. 2020. *KNN Classification*. [online] Available at: <https://www.saedsayad.com/k_nearest_neighbors.htm> [Accessed 31 May 2020].

[8]

Pbs.twimg.com. 2020. [online] Available at: <https://pbs.twimg.com/media/DjnRitgVAAAj9lk.jpg> [Accessed 31 May 2020].