Scientific Computing - Practice sessions

Audio command recognition by DTW and classification

21906085
Duc Thang, NGUYEN
duc-thang.nguyen@univ-tlse3.fr

21902771
Clement, POULL
clement.poull@univ-tlse3.fr

December 7, 2019

ABSTRACT

Speech recognition can be used in many fields, including command recognition for voice drone control. The state of the arts methods for spoken word recognition often base on MFCCs (Mel Frequency Cepstral Coefficients) to extract feature of audio. In this report, we present some approaches to classify the 13 drone commands in French. Firstly, We use dynamic time warping (DTW) algorithm. Second approach, using Principal Component Analysis (PCA) to get feature data reduction of MFCC and then apply k-nearest neighbors (KNN) to classify them. A upgrade of this approach, we combine MFCC and Delta Coefficients to get more features about sound and frame signals change. And finally, we use Convolution neural networks (CNN) into MFCCs directly. The dataset is composed of 235 records without noise and 53 records with noise from 13 native speakers. We use 80% of our dataset to train model, 20% dataset to test and measure the accuracy of approaches. During training, we augment data to variety of situations dataset like pitch, shift, stretch, ignore noise,... The best results base on F1 score, when we use DTW is 83%, PCA + KNN is 91% and CNN is 89%. We also evaluate independently on own dataset (39 files) from native accent and Vietnamese accent with DTW is 79%, PCA + KNN is 95% and CNN is 79%.

Source code: https://github.com/thangdnsf/Audio-command-recognition

Keywords: DTW, PCA, KNN, MFCC, delta coef, CNN, Speech Recognition

1 INTRODUCTION

The pattern speech recognition is rapidly applying in may domains including, health care, military, and telephony,... Which can identify the pattern of content base on the form of wave that reflects physiologic and behavior characteristics from speaker that is called the feature extraction process. There are many feature extraction methods exist for audio but Mel Frequency Cepstral Coefficients (MFCCs) algorithm is a the state of the art method have been widely used. Then they used to classify or measure distance between datas. There are many the state of the art methods to do that such as: Dynamic time warping (DTW), KNN, Neural Network,... In this report, we will implement DTW algorithm, KNN and Convolution Neural networks.

In our case, it will be used to give voice commands for controlling a drone. Unlike direct control methods such as a menu, speech recognition is not trivial and is not always flawless. The computer has no way to directly execute what the speech represents but needs to interpret the voice before execution. This can prove troublesome when the sample has noise, inconsistent speed, volume and many more parameters that can degrade their quality.

As a result, we need a way to accurately interpret the voice commands as one of 13 instructions for controlling a drone: *arretetoi*, *atterrissage*, *avance*, *decollage*, *droite*, *etatdurgence*, *faisunflip*, *gauche*, *plusbas*, *plushaut*, *recule*, *tournedroite*, *tournegauche*.

In the next section will explain about methodology, results, and conclusions. In section II will be explained about the methodology of speech reconciliation process in detail. In the section III will be demonstrate the our results, while section IV show the conclusions.

2 METHODOLOGY

In this section, we will present and explain more detail about the speech recognition process on each approaches. First of all, we present briefly the dataset and take prepossessing data.

2.1 Dataset and prepossessing

The dataset we had access to was composed of sets of the 13 instructions spoken by 13 native speakers in different conditions with audio parameters: 16 KHz, mono, 16 bits, .wav format. The total of dataset include 235 records without noise (by 13 males and 5 females) and 53 records (by 4 males) . During record dataset, Speaker often silent before and after speak word. So the audio file usually has silent two sides of main wave as shown Figure 1. So we need to cut the silent parts of the sample as Figure 2 using trim method in librosa library with threshold to consider as silence is 30.

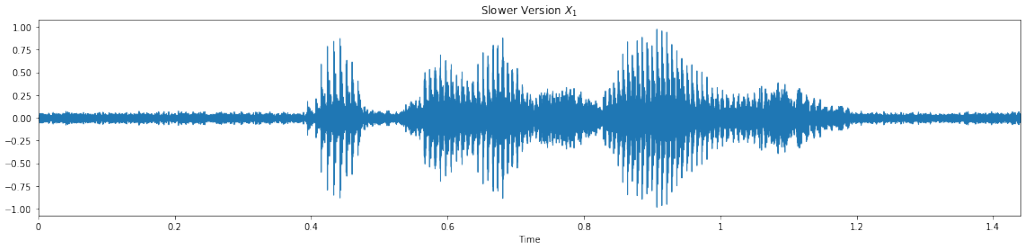


Figure 1 – Sample before trimming.

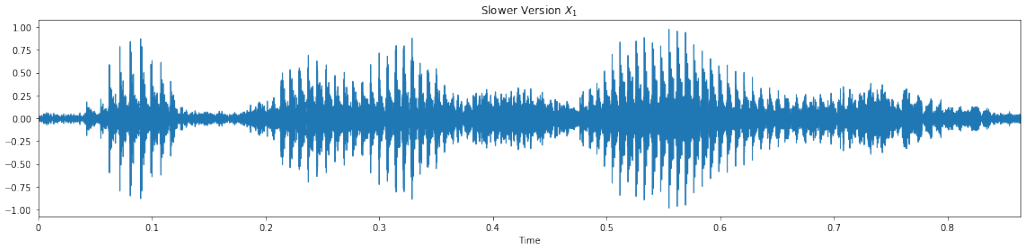


Figure 2 – Sample after trimming.

And the next task needs to reduce noise which affects the quality of audio and accuracy of the model by using noise reduction library based on algorithm by Audacity and get result as Figure 3.

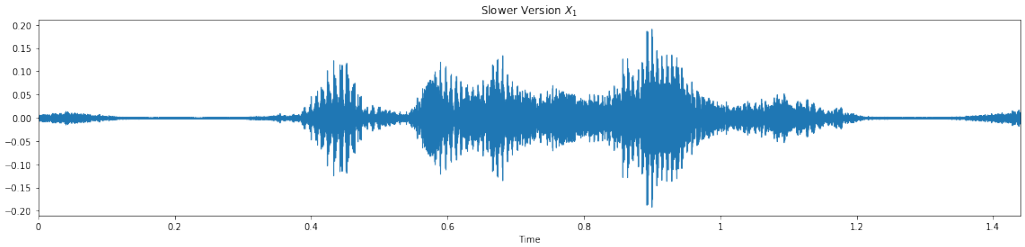


Figure 3 – Sample after remove silent two sides.

We use 80% of our dataset to train model, 20% dataset to test and measure the accuracy of approaches. During training, we has augmented data to variety of situations dataset like pitch, shift, stretch, ignore noise,...

2.2 Feature extraction

After data prepossessing, we will need to extract important features of each audio. One of the sate of the art methods is the mel-frequency cepstral coefficients (MFCCs) algorithm. MFCC is a frequency warping method that allows for a better representation of the response the human earring has, since it does not follow a linear scale. It calculates coefficients by performing a Fourier transform, mapping the result to the mel scale, taking the logs of the powers at certain frequencies, taking the discrete cosine transform of the logs and getting the amplitudes of the resulting spectrum. Figure 4 is a sample of 'etadurgence' word.

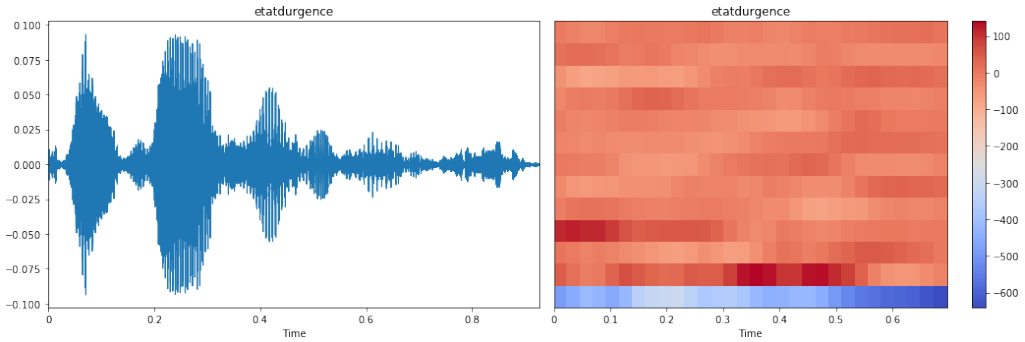


Figure 4 – MFCC of 'etadurgence' word

To get more accuracy, we use Delta Coefficients and Delta Coefficients order 2 to extract feature regarding frame signals change, like the slope of a formant on the transition show example as figure 5.

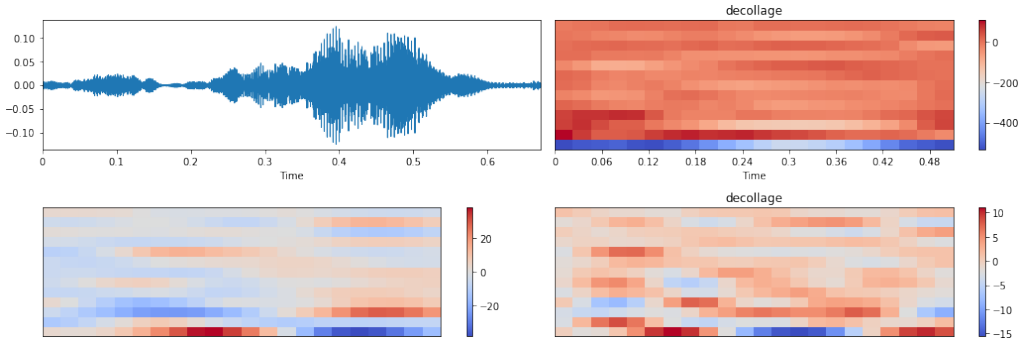


Figure 5 – MFCC, del, del 2 of 'decollage' word

2.3 Methods

2.3.1 Dynamic time warping

The Dynamic time warping (DTW) algorithm is showed in Algorithm1 is a popular method for comparing two sequences and illustrated by Figure 6.

Algorithm 1 DTW algorithm

$g(0,0) \leftarrow 0$
for $j \leftarrow 1$ *to* J **do**
  $g(0,j) \leftarrow +\infty$
for $i \leftarrow 1$ *to* I **do**
  $g(i,0) \leftarrow +\infty$
  for $j \leftarrow 1$ *to* J **do**
    $g(i,j) \leftarrow min(g(i-1,j) + \omega_0 * d(i,j), g(i-1,j-1) + \omega_1 * d(i,j), g(i,j-1) + \omega_2 * d(i,j))$
$D \leftarrow g(I,J)/(I+J)$

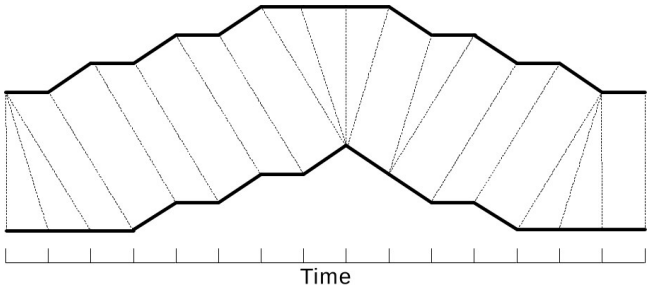


Figure 6 – Warping between two time series[1].

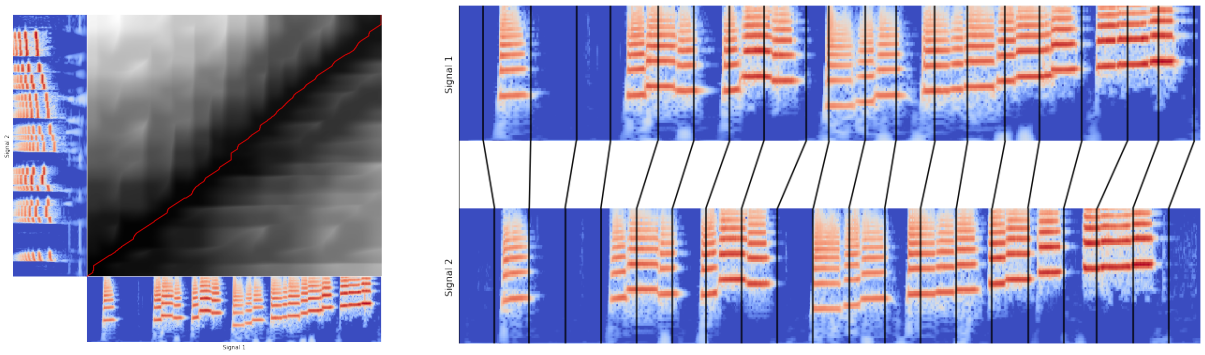


Figure 7 – Compare two MFCCs using DTW[2]

And Figure 7 illustrates the comparison of two MFCCs using DTW (SAKOE; CHIBA, 1978).

The first part our recognition method uses the dynamic time warping (DTW) algorithm on the MFCCs of each samples present in Algorithm 2.

Algorithm 2 MFCCs classification using DTW

```
y_pred = []
for x ← X_test do
    min_dist ← +∞
    min_index = None
    for x_train,y_train ← X_train,Y_train do
        dist ← dtw(x^T, x_train^T)
        if min_dist > dist then
            min_dist ← dist
            min_index = y_train
    y_pred ← min_index
```

2.4 PCA and KNN

The second approach of our recognition method is to apply the principal component analysis (PCA) on our dataset, before classification with the KNN algorithm. PCA is a method that allows to reduce the size of a set by transforming it into a set of linearly uncorrelated variable. It works by calculating the co-variance matrix of the sequence, then by calculating the eigenvalues and eigenvectors of this matrix, choosing a subset of these eigenvectors and projecting the dataset onto a new basis created by the chosen eigenvectors(WINURSITO et al., 2018). Note that, we need to fix size of all data by padding zeros vector to two slide of each MFCC Figure 8 and flatten each MFCC to vector. and Then rescaling our data to a range between -1 and 1, with StandardScaler method from sklearn.

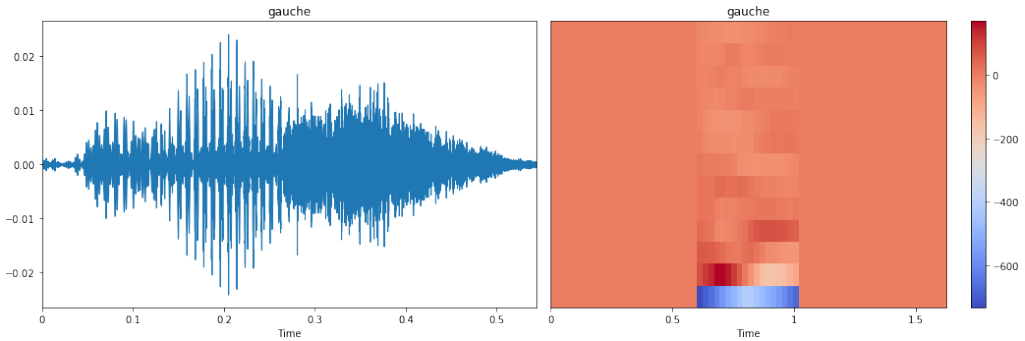


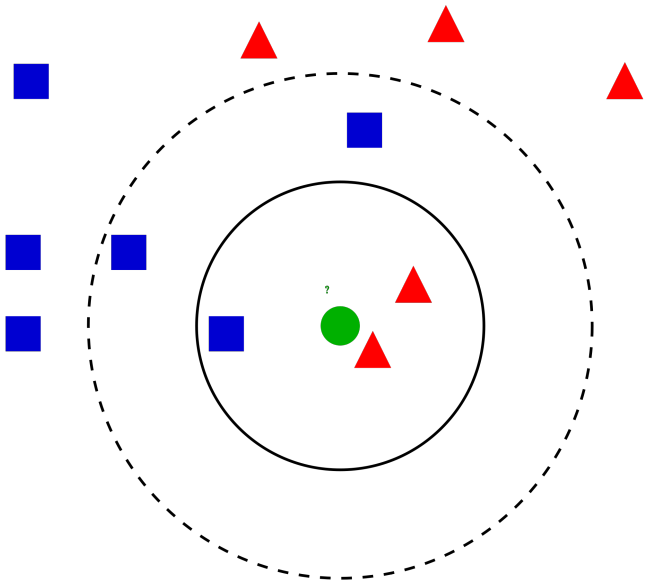
Figure 8 – Fix size MFCC by padding zero vectors

And the we will classify via classification algorithms like KNN, SVM, Neural networks, ... We will perform the K-nearest neighbors (KNN) using KNN method in sklearn library. It works by having a training base, on which is placed a

[1] Florida Institute of Technology
[2] musicinformationretrieval.com

test sample. The training base is composed of vectors with each an associated label, the label on these training vectors represents a class. We then check the neighborhood of the test sample, and take its k closest neighbors. These neighbors are then used to classify the test sample, with techniques such as majority voting, or weighted majority voting.

Figure 9 – KNN with two classes.



2.5 Convolutions neural networks

The convolutions neural networks (CNN) is a neural network that has one or many convolutional layers as Figure10 and are used mainly for image processing, classification, segmentation, . . . And this problem, we apply CNN model with our dataset as a image dataset where a MFCC matrix as a image.

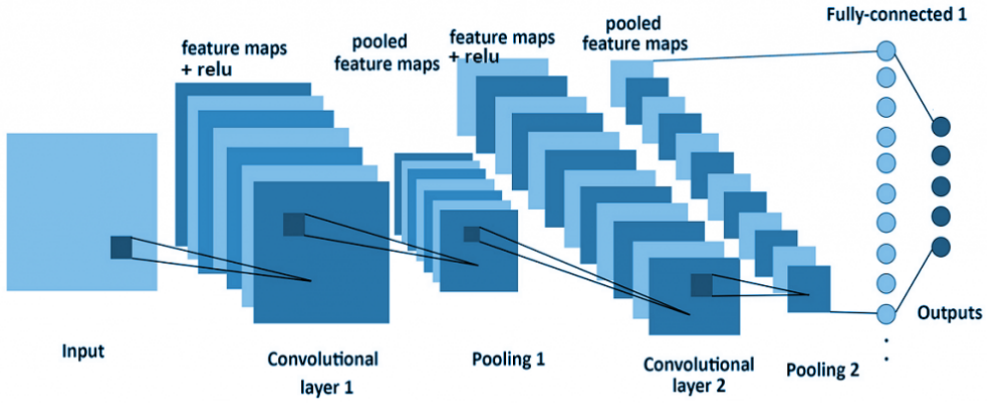


Figure 10 – Convolutions neural networks.

3 PERFORMED TESTS AND RESULTS

In this section, we will implement and present our above proposal methods in more detail. All our experiments are performed in python programming language and using jupyter notebook editor. To evaluate independently, we have recorded own dataset (39 files)by native and Vietnamese accent.

3.1 DTW

We have implemented Algorithm 1 and evaluated by solving exercise TD2 as Figure 11 and 12.

And then, we have evaluated performance of DTW algorithm with the following cases. Firstly, Evaluating influence male/ female voices and we have got so good result as Figure 13. Secondly, Evaluating influence background noises on recognition (learning set is data without noise and test set is data with noise) and we have got bad result as Figure 14. Thirdly, Evaluating with normal (random noise and without noise) data as Figure 15. And finally, evaluating ability to new voice as Figure 16

distant= 3.6

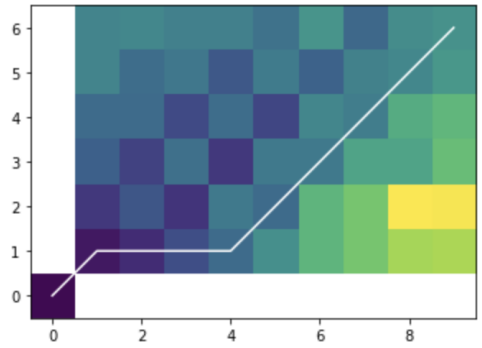


Figure 11 – Comparing two numerical sequences

distant= 0.16666666666666666

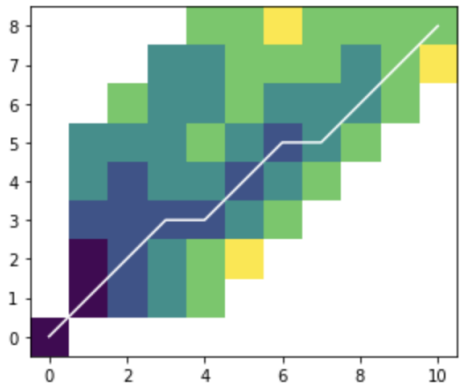
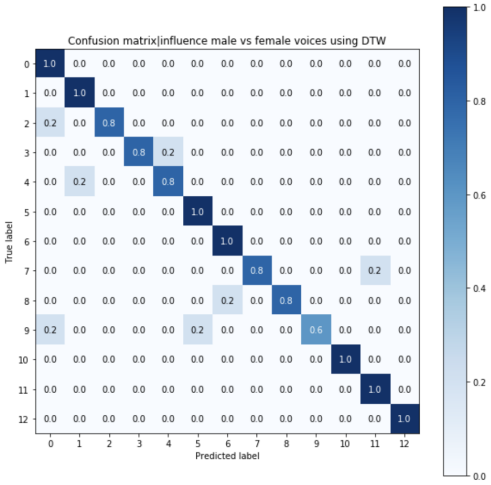
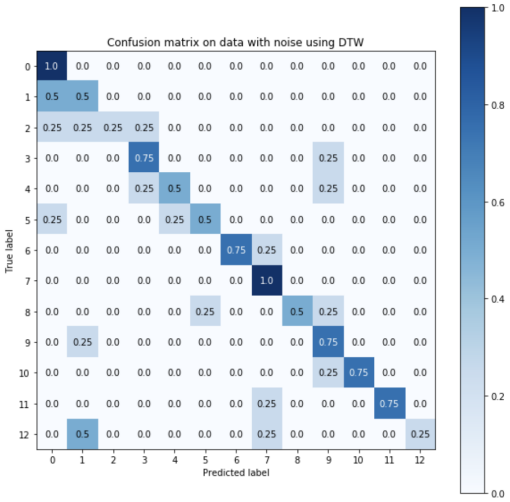


Figure 12 – Comparing DNA sequences



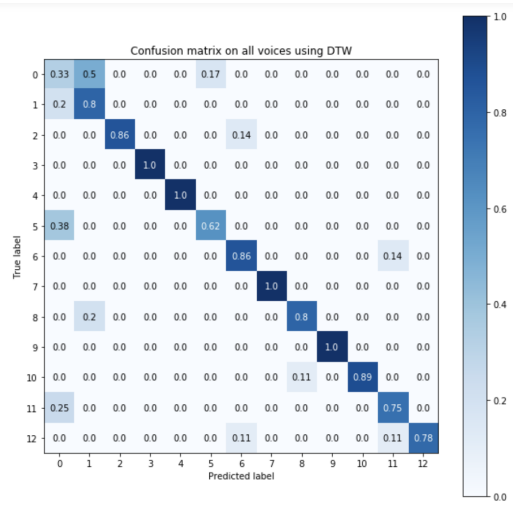
	precision	recall	f1-score	support
0	0.71	1.00	0.83	5
1	0.83	1.00	0.91	5
2	1.00	0.80	0.89	5
3	1.00	0.80	0.89	5
4	0.80	0.80	0.80	5
5	0.83	1.00	0.91	5
6	0.83	1.00	0.91	5
7	1.00	0.80	0.89	5
8	1.00	0.80	0.89	5
9	1.00	0.60	0.75	5
10	1.00	1.00	1.00	5
11	0.83	1.00	0.91	5
12	1.00	1.00	1.00	5
accuracy			0.89	65
macro avg	0.91	0.89	0.89	65
weighted avg	0.91	0.89	0.89	65

Figure 13 – Evaluating DTW algorithm influence male/ female voices



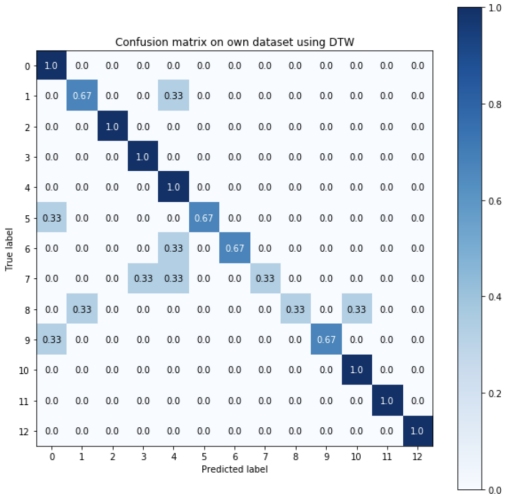
	precision	recall	f1-score	support
0	0.50	1.00	0.67	4
1	0.33	0.50	0.40	4
2	1.00	0.25	0.40	4
3	0.60	0.75	0.67	4
4	0.67	0.50	0.57	4
5	0.67	0.50	0.57	4
6	1.00	0.75	0.86	4
7	0.57	1.00	0.73	4
8	1.00	0.50	0.67	4
9	0.43	0.75	0.55	4
10	1.00	0.75	0.86	4
11	1.00	0.75	0.86	4
12	1.00	0.25	0.40	4
accuracy			0.63	52
macro avg	0.75	0.63	0.63	52
weighted avg	0.75	0.63	0.63	52

Figure 14 – Evaluating DTW algorithm influence background noises on recognition



	precision	recall	f1-score	support
0	0.29	0.33	0.31	6
1	0.50	0.80	0.62	5
2	1.00	0.86	0.92	7
3	1.00	1.00	1.00	6
4	1.00	1.00	1.00	8
5	0.83	0.62	0.71	8
6	0.75	0.86	0.80	7
7	1.00	1.00	1.00	4
8	0.80	0.80	0.80	5
9	1.00	1.00	1.00	8
10	1.00	0.89	0.94	9
11	0.60	0.75	0.67	4
12	1.00	0.78	0.88	9
accuracy			0.83	86
macro avg	0.83	0.82	0.82	86
weighted avg	0.86	0.83	0.83	86

Figure 15 – Evaluating DTW algorithm with normal (random noise and without noise) data



	precision	recall	f1-score	support
0	0.60	1.00	0.75	3
1	0.67	0.67	0.67	3
2	1.00	1.00	1.00	3
3	0.75	1.00	0.86	3
4	0.50	1.00	0.67	3
5	1.00	0.67	0.80	3
6	1.00	0.67	0.80	3
7	1.00	0.33	0.50	3
8	1.00	0.33	0.50	3
9	1.00	0.67	0.80	3
10	0.75	1.00	0.86	3
11	1.00	1.00	1.00	3
12	1.00	1.00	1.00	3
accuracy			0.79	39
macro avg	0.87	0.79	0.78	39
weighted avg	0.87	0.79	0.78	39

Figure 16 – Evaluating DTW algorithm ability to new voice

3.2 PCA and KNN

We have augmented dataset by changing pitch with Frequency (+15%,+25%,+50%,-15%,-25%,-50%). Which permits dataset is better make our model to do not depend on accent and reduce noise. In here, we implement following upgrade version: using delta and delta 2 of MFCC as Figure 17(WINURSITO et al., 2018).

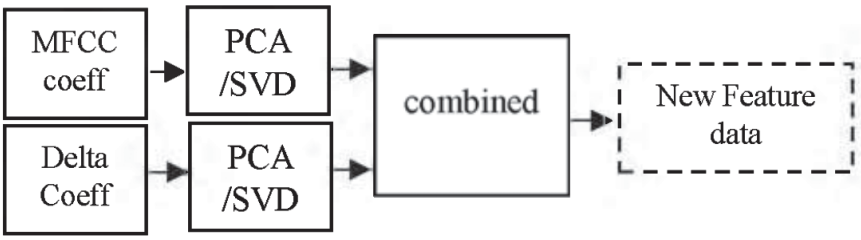
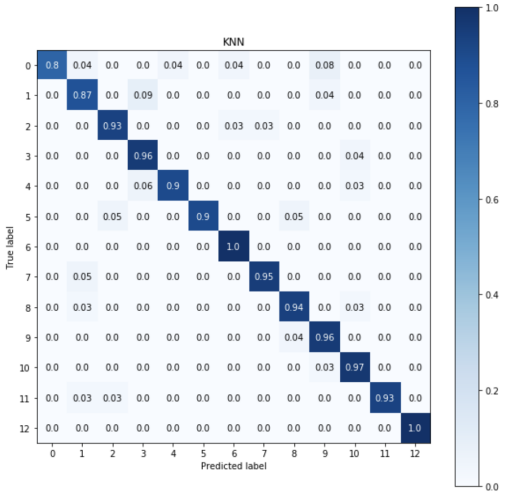


Figure 17 – The MFCC + Del + Del2 +PCA + KNN process.

And we have got best results on test data is 93% as Figure 18 and 95% on own data as Figure 19.

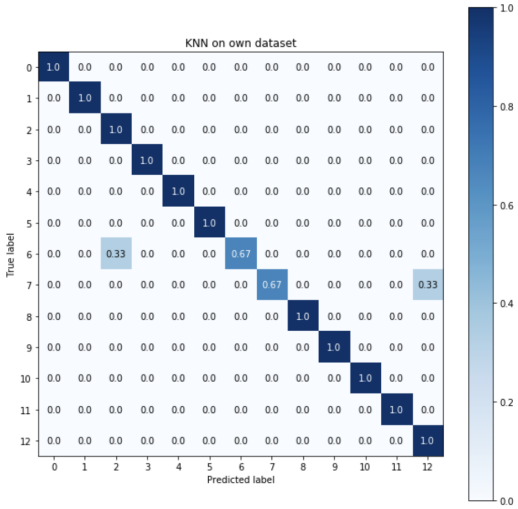
3.3 CNN

We have tried to train Deep Neural network model - CNN. Figure 20 present architecture CNN model with 14,301 parameters. And Figure 21 and 22 show the loss and accuracy after 6000 epochs.



	precision	recall	f1-score	support
0	1.00	0.80	0.89	25
1	0.83	0.87	0.85	23
2	0.93	0.93	0.93	29
3	0.85	0.96	0.90	23
4	0.97	0.90	0.93	31
5	1.00	0.90	0.95	21
6	0.93	1.00	0.97	28
7	0.95	0.95	0.95	22
8	0.94	0.94	0.94	31
9	0.85	0.96	0.90	24
10	0.91	0.97	0.94	31
11	1.00	0.93	0.96	29
12	1.00	1.00	1.00	21
accuracy			0.93	338
macro avg	0.94	0.93	0.93	338
weighted avg	0.94	0.93	0.93	338

Figure 18 – Evaluating PCA+KNN model normal (random noise and without noise) data



	precision	recall	f1-score	support
0	1.00	1.00	1.00	3
1	1.00	1.00	1.00	3
2	0.75	1.00	0.86	3
3	1.00	1.00	1.00	3
4	1.00	1.00	1.00	3
5	1.00	1.00	1.00	3
6	1.00	0.67	0.80	3
7	1.00	0.67	0.80	3
8	1.00	1.00	1.00	3
9	1.00	1.00	1.00	3
10	1.00	1.00	1.00	3
11	1.00	1.00	1.00	3
12	0.75	1.00	0.86	3
accuracy			0.95	39
macro avg	0.96	0.95	0.95	39
weighted avg	0.96	0.95	0.95	39

Figure 19 – Evaluating PCA+KNN ability to new voice

```
Model: "sequential_3"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_3 (Conv2D)            (None, 12, 59, 32)        160
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 11, 58, 48)        6192
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 5, 29, 48)         0
_________________________________________________________________
dropout_3 (Dropout)          (None, 5, 29, 48)         0
_________________________________________________________________
avg_pool (GlobalAveragePooli (None, 48)                0
_________________________________________________________________
dense_2 (Dense)              (None, 128)               6272
_________________________________________________________________
dropout_4 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_3 (Dense)              (None, 13)                1677
=================================================================
Total params: 14,301
Trainable params: 14,301
Non-trainable params: 0
_________________________________________________________________
```

Figure 20 – The CNN architecture model

And we have got good results on test data is 91% as Figure 23 and 79% on own data as Figure 24.

4 CONCLUSIONS

The primary objective, we have successfully implemented DTW algorithm, PCA, KNN and CNN methods and get good results. We see that, data augmentation make better accuracy and combining MFCC + del + del2 and PCA + KNN has

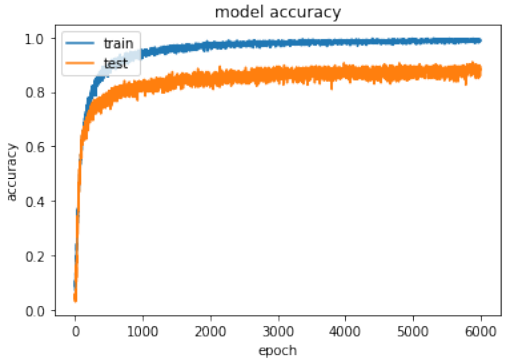


Figure 21 – The loss and accuracy during training model.

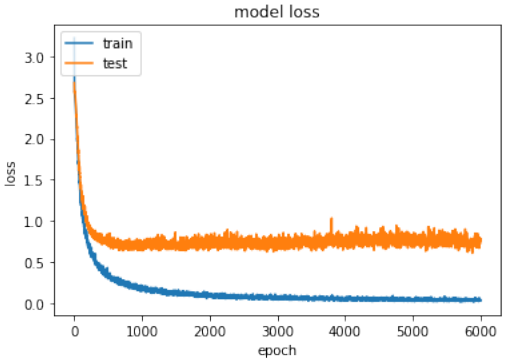
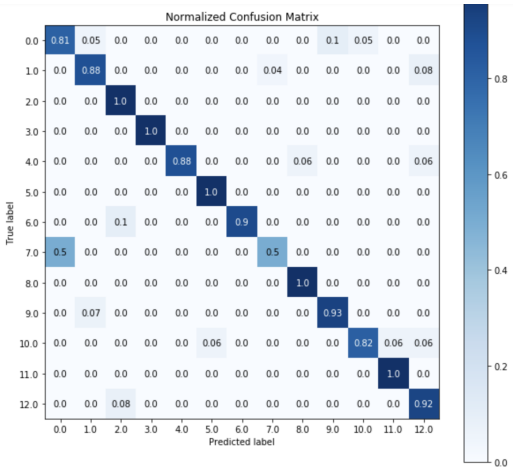
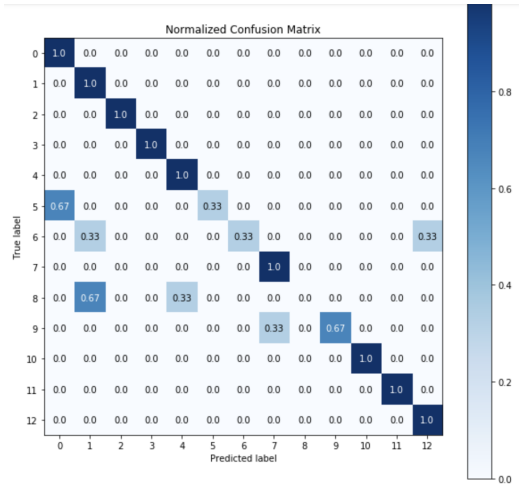


Figure 22 – The loss and accuracy of test set.



	precision	recall	f1-score	support
0.0	0.77	0.81	0.79	21
1.0	0.92	0.88	0.90	26
2.0	0.56	1.00	0.71	5
3.0	1.00	1.00	1.00	22
4.0	1.00	0.88	0.94	17
5.0	0.91	1.00	0.95	10
6.0	1.00	0.90	0.95	20
7.0	0.83	0.50	0.62	10
8.0	0.96	1.00	0.98	26
9.0	0.88	0.93	0.90	15
10.0	0.93	0.82	0.87	17
11.0	0.97	1.00	0.98	31
12.0	0.86	0.92	0.89	26
accuracy			0.91	246
macro avg	0.89	0.90	0.88	246
weighted avg	0.92	0.91	0.91	246

Figure 23 – Evaluating CNN model normal (random noise and without noise) data



	precision	recall	f1-score	support
0	0.60	1.00	0.75	3
1	0.50	1.00	0.67	3
2	1.00	1.00	1.00	3
3	1.00	1.00	1.00	3
4	0.75	1.00	0.86	3
5	1.00	0.33	0.50	3
6	1.00	0.33	0.50	3
7	0.75	1.00	0.86	3
8	0.00	0.00	0.00	3
9	1.00	0.67	0.80	3
10	1.00	1.00	1.00	3
11	1.00	1.00	1.00	3
12	0.75	1.00	0.86	3
accuracy			0.79	39
macro avg	0.80	0.79	0.75	39
weighted avg	0.80	0.79	0.75	39

Figure 24 – Evaluating CNN ability to new voice

best results. Although, using DTW on data augmentation is more expensive, I think it's a simple method and good result.

REFERENCES

SAKOE, H.; CHIBA, S. Dynamic programming algorithm optimization for spoken word recognition. **IEEE Transactions on Acoustics, Speech, and Signal Processing**, v. 26, n. 1, p. 43–49, Feb. 1978. ISSN 0096-3518. DOI: 10.1109/TASSP.1978.1163055.

WINURSITO, A. et al. Feature Data Reduction of MFCC Using PCA and SVD in Speech Recognition System. In: 2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE). [S.l.: s.n.], July 2018. p. 1–6. DOI: 10.1109/ICSCEE.2018.8538414.