

Implementing The Speaker Recognition Process In Python Using 'Thuyg-20-Sre' Data Set

Rajat Verma

Harshita Mishra

Dr.Namrata Dhanda

Department of Computer Science & Engineering,
Amity School of Engineering & Technology, Amity University, Lucknow

Abstract: *the process of Speaker Recognition has been used in various domains by the researchers. This can be for a biometric system or for entertainment purposes. There is a difference between the process of speaker recognition and speech recognition. A large number of datasets are available on the internet but the dataset that I have used is 'THUYG-20-SRE'. The components of this dataset is also mentioned in this paper. A brief difference between the functionality when the dataset is present or absent is also mentioned in this paper. The use of MFCC and delta for the audio extraction is also mentioned in this paper.*

Keywords: *Speaker, Recognition, Speech, Dataset, Speaker Feature, Training, Testing.*

I. INTRODUCTION TO SPEAKER RECOGNITION

When one has to communicate to a machine, the most efficient manner is the usage of speech. Training a machine is also an effective mechanism when done by the medium of speech. Voice Biometrics mechanism plays the main role in the identification of the speaker that includes the features of voices. Another synonym of this procedure is voice recognition. There is a border line between the terms Speaker Recognition and Speech Recognition. They are not the same! The Speaker Recognition deals with the recognition procedure that "who" is speaking and the speech recognition deals with the recognition of "what" is being said. The knowledge of language is an essential component for the Accuracy of Recognition. Humans use a large amount of techniques to remove the confusions in what they hear, Speaker Recognizers should use this technique too in order to make themselves correct. Language structure also plays a great role in the same scenario of Speaker Recognition. Speaker Recognition may vary on the dependent characteristics of the speaker. The Recognition of the patterns is a super class of the speaker's recognition problem. Voice Print can be processed using

various tools and mechanisms such as Hidden Markov Model commonly abbreviated as HMM, Gaussian Mixture Models, Neural networks (NN) etc.

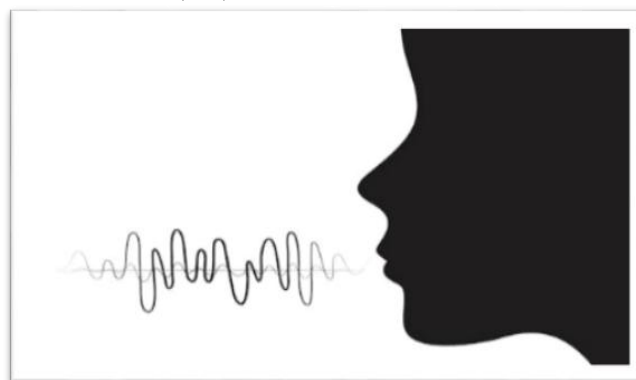


Figure 1

Where the term of "Computational Linguistics" [3] is concerned, the speech recognition acts as a subfield and does the work of converting the words into text matter. Speaker recognition can be done with the help of datasets also. An

example of this can be “THUYG-20-SRE” [1]. It is an open as well as free database.

II. PROCESS OF SPEAKER RECOGNITION WHEN DATASET IS NOT PRESENT

The process of Speaker Recognition [2] is typically done in 2 phases:

- ✓ Enrollment Procedure
- ✓ Verification Procedure

In the initial phase, the voices of the users are included and the enormous features are extracted to construct a model or a hypothesis. In the later phase, the voice prints that were created in the enrolling phase are compared to the samples of speech.

The Enrollment procedure [4] contains following elements if a dataset is not pre-configured:

- ✓ Creation of profile
- ✓ Deletion of profile
- ✓ Response of Enrollment
- ✓ Enrollment of Profile
- ✓ Get Profile phase
- ✓ Identification profile
- ✓ Identification Response
- ✓ Identification Service Http Client Helper
- ✓ Identify Files
- ✓ Print All Profiles
- ✓ Profile Creation Response
- ✓ Reset Enrollments

The Verification Phase [5] has the following components if a data set is not preconfigured:

- ✓ Creation of profile
- ✓ Deletion of profile
- ✓ Response of Enrollment
- ✓ Enrollment of profile
- ✓ Get Profiles
- ✓ Print All Profiles
- ✓ Profile Creation Response
- ✓ Reset Enrollments
- ✓ Verification Profile
- ✓ Verification Response
- ✓ Verification Service Http Client Helper
- ✓ Verify File

The creation of profiles is a must in situation where data set is not present. Enrolling profiles may take a number of times for an assurance that the user is valid. The Cognitive Services of Microsoft usually does this 3 times for increasing the self-confidence of the machine.

The Deletion of profiles is also an important step in the speaker recognition as in case of checking the validity of the user. The subscription is legal or not can be identified in this case.

Once the speaker is enrolled, the audio sample can be entered so that it can be compared to the enrolled voice, identifying the original user.

When the need is over, the enrollments can be reset and new entries can be entered and the whole step can be repeated. Coming to the verification phase, the steps that were done in the identification are similar to the verification scenario

naming phase but a bit different in coding phase. The similar naming phases includes:

- ✓ Creation of profile
- ✓ Deletion of profile
- ✓ Response of Enrollment
- ✓ Enrollment of profile
- ✓ Get Profiles
- ✓ Print All Profiles
- ✓ Profile Creation Response
- ✓ Reset Enrollments

The printing of profiles is a convenient objective that is fulfilled every time whenever documentation is a pre-requisite. The response can be positive or negative depending on the matching of the voice prints enrolled previously. If the result is positive means the confidence is high that means the user is valid as well as authorized, and if not that means negative then in that condition the user is considered as faulty.

The Response of the enrollment can be depicted with the encapsulation of enrollment response. Encapsulation is a process of binding of data and function into a single unit. One example of the encapsulation procedure is there are many departments in a factory, now a person in the sales department has to meet a person in registration department and for that the sales person has to issue a memo for that, this illustrates the use of encapsulation.

The Enrollment has to be done on a server also. The identification of the profile has a class that encapsulates a user profile.

Major researches have seen various models in which the most popular was Gaussian Mixture Model [6].

Three approaches that are mostly followed:

- ✓ Pitch [7]
- ✓ Segmentation[8]: Implicit- Unsupervised Clustering [9]
- ✓ Explicit- Hidden Markov Model [10]

NEURAL NETWORKS [11]

- ✓ Pitch: It is a property of sound, basically concern to frequency. It can be termed as the quality of sound that may be depicted as high or low, considering a person to have a sharp voice or heavy voice.
- ✓ Unsupervised Clustering: Target attribute is unavailable. In this no teacher is present. The main objective is to form groups and the properties of elements of a particular group are similar to each other whereas the elements of different clusters are having the different properties. Clustering is always compared to classification procedure. This procedure deals with classification and is more organized in comparison to clustering. Various types of clustering includes hierarchical that is further divided into 2 categories: agglomerative and divisive, partition that includes k-means, k-medoid etc.
- ✓ Hidden Markov Model: This model is similar to the Markov model, the characteristic that distinguishes it with the renowned model is “hidden states”. Let’s take an example, a monkey and a man were talking softly behind a curtain, one can say whispering, another man standing at a good distance listens to them, but he can’t distinguish that who is man and who is monkey. The whispering is termed as the visible states where as the

monkey as well as the man are hidden! This is used for forecasting purposes also and the major popular example of this is of two coins.

- ✓ Neural Networks: In concern to biology, the largest cell is “neuron” that performs millions of operations and its components like dendrites, axon etc. plays a really vital role. In terms in artificial neural network the word “perceptron” is a measure of observation.

Processing elements are interconnected with each other and performs a gradient descent methods. These processing components is considered as “neurons”. The skilled or the competent neural networks allows the interoperability of the network that has the learning ability. In case of realization of logic gates also, learning is applied to cope up with the manual solution (hit and trial) as it was in Mc Culloch Pitts Model.

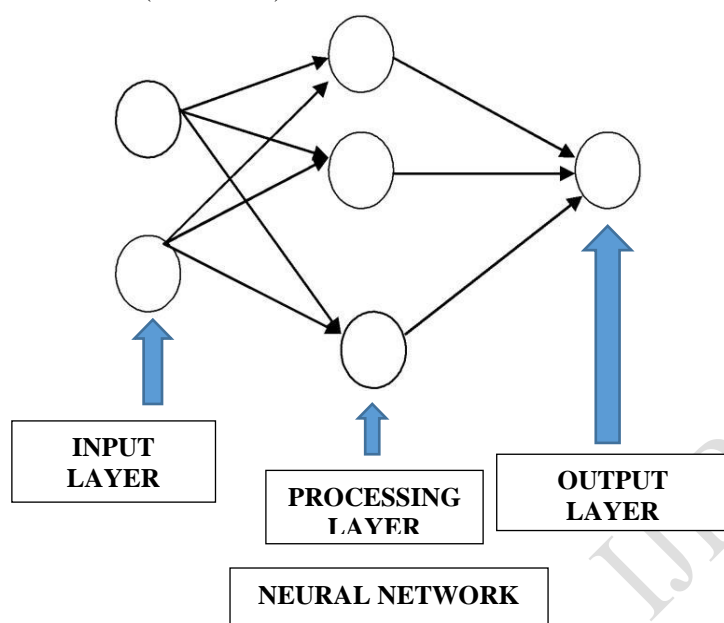


Figure 2

For the purpose of pattern recognition and classification neural networks are essentially used. Realization of logic gates can also be done using the neural networks. An example of it was McCulloch Pitts [12]. Minimization of error takes place when weights are adjusted in a nominal fashion. Backpropagation algorithm performs a vital role in finding the gradient of the error. In the phase of acquiring new information and problem solving, the processed data that go through the network corresponds to the weight adjustment scenario. In the multiclass scenario of neural network abbreviated as NN, the difficulty can be marked by making use of the feed forward technique in multi-layer division, where a single neuron is not used. The other types of neuron are as single layer feed forward and feedback networks.

III. PROCESS OF SPEAKER RECOGNITION WHEN DATASET IS PRESENT-‘ THUYG-20-SRE’

THUYG-20-SRE is a very popular data set. The total size of the dataset is about 6.5 GB. The Components of the data set is mentioned in the table given below-

NAME	SIZE	DESCRIPTION
<u>data_thuyg20.tar.gz</u>	2.1 GB	speech data and transcripts for speech recognition
<u>data_thuyg20_sre.tar.gz</u>	1.6 GB	speech data for speaker recognition
<u>test_noise.tar.gz</u>	773 MB	standard Odb noisy test data for speech recognition
<u>test_noise_sre.tar.gz</u>	1.9 GB	standard Odb noisy test data for speaker recognition
<u>resource.tar.gz</u>	26 MB	supplementary resources, incl. lexicon for training data, noise samples

Table 1

The dataset can be freely downloaded from the openslr.org. The process of speaker recognition takes place in three steps-

- ✓ Speaker Features [13]
- ✓ Train Model [14]
- ✓ Test Speaker [15]

In the initial step, the speaker features are taken then the model or the hypothesis is trained and for that a text file containing the names of the speakers is also required. In the final step the speakers are tested and verified. They are modelled in a group of 5. When the model is trained then, .gmm files are generated whose path is to be set in the test speaker python file. The MFCC [16] is an abbreviation of Mel Frequency Cepstral Coefficients present in python_speech_features.mfcc (). The full form of GMM [17] is Group Mail Message Log File. Log Filter Energies are also used in this process present in python_speech_features.logfbank ().

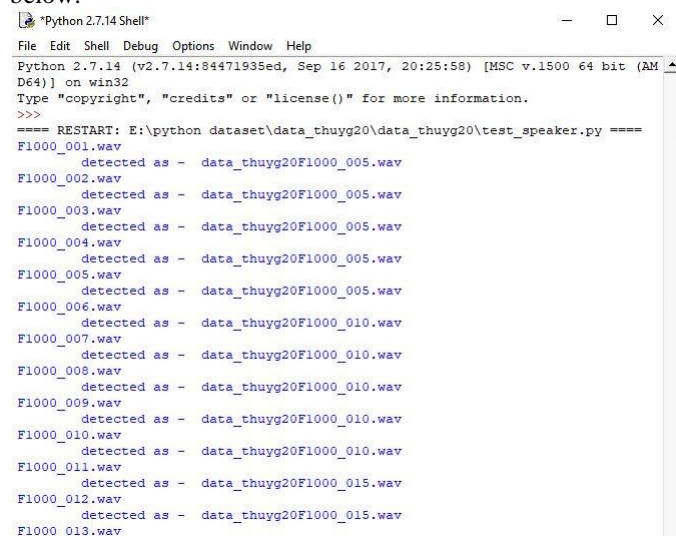
A number of dim MFCC is used with some of frame log energy. A number of dim delta computation is done on MFCC features. The coding is utf-8 [18].

As output, it returns dimensional features vectors for an audio [19].

The sample of the output of the training model is depicted in the figure given below:

```
Python 2.7.14 Shell
File Edit Shell Debug Options Window Help
Python 2.7.14 (v2.7.14:84471935ed, Sep 16 2017, 20:25:58) [MSC.v.1500 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
==== RESTART: E:\python dataset\data_thuyg20\data_thuyg20\train_models.py ====
F1000_001.wav
F1000_002.wav
F1000_003.wav
F1000_004.wav
F1000_005.wav
+ modeling completed for speaker: F1000_005.wav.gmm with data point = (3247L, 40L)
F1000_006.wav
F1000_007.wav
F1000_008.wav
F1000_009.wav
F1000_010.wav
+ modeling completed for speaker: F1000_010.wav.gmm with data point = (3350L, 40L)
F1000_011.wav
F1000_012.wav
F1000_013.wav
F1000_014.wav
F1000_015.wav
```

The sample of the output of the testing of speaker is given below:



```
Python 2.7.14 (v2.7.14:84471935ed, Sep 16 2017, 20:25:58) [MSC v.1500 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
==== RESTART: E:\python dataset\data_thuyg20\data_thuyg20\test_speaker.py ====
F1000_001.wav detected as - data_thuyg20F1000_005.wav
F1000_002.wav detected as - data_thuyg20F1000_005.wav
F1000_003.wav detected as - data_thuyg20F1000_005.wav
F1000_004.wav detected as - data_thuyg20F1000_005.wav
F1000_005.wav detected as - data_thuyg20F1000_005.wav
F1000_006.wav detected as - data_thuyg20F1000_010.wav
F1000_007.wav detected as - data_thuyg20F1000_010.wav
F1000_008.wav detected as - data_thuyg20F1000_010.wav
F1000_009.wav detected as - data_thuyg20F1000_010.wav
F1000_010.wav detected as - data_thuyg20F1000_010.wav
F1000_011.wav detected as - data_thuyg20F1000_015.wav
F1000_012.wav detected as - data_thuyg20F1000_015.wav
F1000_013.wav
```

The dataset contains an enormous amount of entries, so a sample is appropriate to depict the entire illustration.

IV. CONCLUSION

The Speaker Recognition Scenario has a tremendous amount of potential [20] in providing support to applications in various domains. This paper initially depicts and illustrates the use of speaker recognition in this modern era. After that, the process of speaker recognition is illustrated that includes 2 processes that are enrollment and recognition. The difference between the speaker recognition and speech recognition is also illustrated. When the dataset is present, then in that case how the speakers are identified is also illustrated. In future emotions can also be added to an accurate rate and making this speaker recognition system more valuable. They have the capability to enhance the security to a larger extent.

REFERENCES

- [1] Rozi, A., Wang, D., Zhang, Z., & Zheng, T. F. (2015, October). An open/free database and Benchmark for Uyghur speaker recognition. In Oriental COCODA held jointly with 2015 Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE), 2015 International Conference (pp. 81-85). IEEE.
- [2] Gerl, F., & Herbig, T. (2009). U.S. Patent Application No. 12/249,089.
- [3] Church, K. W., & Mercer, R. L. (1993). Introduction to the special issue on computational linguistics using large corpora. *Computational linguistics*, 19(1), 1-24.
- [4] Reynolds, D. A. (2002, May). An overview of automatic speaker recognition technology. In *Acoustics, speech, and signal processing (ICASSP), 2002 IEEE international conference on* (Vol. 4, pp. IV-4072). IEEE.
- [5] Reynolds, D. A. (1995). Speaker identification and verification using Gaussian mixture speaker models. *Speech communication*, 17(1-2), 91-108.
- [6] Reynolds, D. A., Quatieri, T. F., & Dunn, R. B. (2000). Speaker verification using adapted Gaussian mixture models. *Digital signal processing*, 10(1-3), 19-41.
- [7] Zhu, J. W., Sun, S. F., Liu, X. L., & Lei, B. J. (2009, August). Pitch in speaker recognition. In *Hybrid Intelligent Systems, 2009. HIS'09. Ninth International Conference on* (Vol. 1, pp. 33-36). IEEE..
- [8] Foote, J. T., & Wilcox, L. (2002). U.S. Patent No. 6,404,925. Washington, DC: U.S. Patent and Trademark Office.
- [9] Shum, S. H., Reynolds, D. A., Garcia-Romero, D., & McCree, A. (2014). Unsupervised clustering approaches for domain adaptation in speaker recognition systems.
- [10] Lee, K. F., & Hon, H. W. (1989). Speaker-independent phone recognition using hidden Markov models. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(11), 1641-1648.
- [11] Lei, Y., Scheffer, N., Ferrer, L., & McLaren, M. (2014, May). A novel scheme for speaker recognition using a phonetically-aware deep neural network. In *Acoustics, Speech and Signal*
- [12] Processing (ICASSP), 2014 IEEE International Conference on(pp. 1695-1699). IEEE.
- [13] Nava, P. A., & Taylor, J. M. (1996, September). Speaker independent voice recognition with a fuzzy neural network. In *Fuzzy Systems, 1996., Proceedings of the Fifth IEEE International Conference on* (Vol. 3, pp. 2049-2052). IEEE.
- [14] Soong, F. K., Rosenberg, A. E., Juang, B. H., & Rabiner, L. R. (1987). Report: A vector quantization approach to speaker recognition. *Bell Labs Technical Journal*, 66(2), 14-26.
- [15] Lee, L. M., & Lee, J. C. (2006, June). A study on high-order hidden Markov models and applications to speech recognition. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems* (pp. 682-690). Springer, Berlin, Heidelberg.
- [16] Doddington, G., Liggett, W., Martin, A., Przybocki, M., & Reynolds, D. (1998). Sheep, goats, lambs and wolves: A statistical analysis of speaker performance in the NIST 1998 speaker recognition evaluation. *NATIONAL INST OF STANDARDS AND TECHNOLOGY GAITHERSBURG MD*.
- [17] Murty, K. S. R., & Yegnanarayana, B. (2006). Combining evidence from residual phase and MFCC features for speaker recognition. *IEEE signal processing letters*, 13(1), 52-55.
- [18] Astarabadi, S. (1998). U.S. Patent No. 5,822,405. Washington, DC: U.S. Patent and Trademark Office.
- [19] Scannell, K. P. (2007, September). The Crúbadán Project: Corpus building for under-resourced languages. In *Building and Exploring Web Corpora: Proceedings of the 3rd Web as Corpus Workshop* (Vol. 4, pp. 5-15).
- [20] Campbell, W. M., Sturim, D. E., & Reynolds, D. A. (2006). Support vector machines using GMM supervectors for speaker verification. *IEEE signal processing letters*, 13(5), 308-311.
- [21] Doddington, G. R., Przybocki, M. A., Martin, A. F., & Reynolds, D. A. (2000). The NIST speaker recognition

evaluation—overview, methodology, systems, results, perspective. Speech Communication, 31(2), 225-254.

IJIRAS