The Application of Machine Learning InSpeech Emotion Recognition

Dongkun Xu [1, *]

[1] Department of Engineering, University of Nottingham, Nottingham, United Kingdom

* Corresponding Author Email: efydx1@nottingham.ac.uk

Abstract. In many applications, such as voice assistants, call centers, psychological counseling, business negotiation, and even hostage rescue, it is becoming more and more important to know the mental state of the other side of the conversation. This article introduces a Speech emotion recognition project based on python and Librosa. This project uses machine learning to train computers to judge the emotional state of a speaker from human speech. Based on this project, the accuracy and efficiency of different models as well as potential application directions are discussed. This project uses an artificial neural network (ANN) model to train the data through a multi-layer classifier (MLP). The ANN model used in this project and the influence of different parameters in the model are discussed, and higher accuracy is obtained on the basis of existing models and data. The article will focus on the analysis of the model structure and the effects of different parameters in the model and their corresponding optimal intervals.

Keywords: artificial neuron network, machine learning, speech emotion recognition.

1. Introduction

The ability to perceive the feelings of others is very common among people. In the process of communication with the call center or customer service center [1], the ability to recognize other people's emotions is vividly reflected. The experienced service personnel will always better understand the customer's mood and make appropriate responses. The way of communicating that considers the emotions of others is obviously more beneficial to the service.

As people increasingly communicate with voice assistants, the "intelligence" of voice assistants is also increasingly required. The reason why many people think those voice assistants are not 'intelligent' is that they tend to respond only based on the user's language content, often ignoring the current context and the user's tone of voice [2]. Due to the diversity of human emotions and communication methods, the same language often shows different or even opposite meanings in different tones. This makes it impossible for the voice assistant to always respond to the user's communication with a 'human touch’.

From the data of the World Health Organization [3], it can be found that there are a large number of people with mental disorders in the world. Mental illnesses often bring pain and trouble to patients and even society. The most effective way to relieve mental illness is psychological counseling.

Mental health counseling is an important yet in-demand profession today.

First of all, the shortage of mental health counselors stems from the difficulty of training the related talents and the long learning cycle. Especially in underdeveloped areas of the world, it is often very difficult to train a qualified psychological counselor [3].

Second, the outbreak of Covid-19 has also brought new challenges. The continuous increase in the number of patients has made the mental health situation in many regions less optimistic [4]. The online medical treatment makes it impossible for doctors and patients to communicate face-to-face, which also brings more difficulty in the diagnosis for doctors.

Terrorism, extremism, territorial conflicts, etc. often put innocent people at risk. Unfortunately, the world still experiences many worrying incidents such as terrorist attacks and hostage kidnappings every year [5,6]. Negotiation is an indispensable part of solving such problems [7].

If voice assistants can understand people's tone of voice and identify the user's mood through voice, more 'humane' voice assistants and artificial intelligence communication software will be realized.

If doctors can know the patient's mood in a more efficient and accurate way, it will greatly reduce the workload of medical staff and the pressure on medical staff to study. This will undoubtedly improve the mental health of people in many underdeveloped areas.

If in the process of rescuing a hostage or negotiating with terrorism, the emotions of the other party can be understood. Even based on this to predict the possible future behavior of terrorists. The success rate of negotiations will be greatly improved.

Speech emotion recognition (SER) is a new technology taking advantage of ML. It allows computers to identify human emotional states using data such as tones and pitch in human language. Speech emotion recognition has grown from a small market to an important part of Human-Computer Interaction (HCI) [8, 9] with many applications.

However, speech emotion recognition has never been an easy task for computers. Speech emotion recognition faces many difficulties. Firstly, human speech features are diverse, it includes speech rate, intonation, loudness, style, etc. Some traits were significantly associated with emotion, while others were not. Secondly, human emotions are not simple 0 or 1. People often have multiple emotions at the same time. Complex emotions often appear, and many emotions are very similar. Finally, human emotions are not static. Emotions are constantly changing. In a short period of time, even within a sentence, the speaker's emotions may also change.

This project research uses the RAVDESS dataset [10] for model training and testing. Speech emotion recognition is performed using artificial neural network ANN models, respectively. After experimenting and analyzing the relevant parameters in the model one by one, a theoretically more optimized training model is obtained under the premise of balancing time cost, and accuracy.

2. Methodology

2.1. Experiment process

The main process of this study is to first analyze the input audio signal (training data) and extract audio features, and then use machine learning technology to classify it according to the speaker's mood. After training, the SER model can be obtained, as shown in Figure 1. Finally, use the test set to test to get the model accuracy.

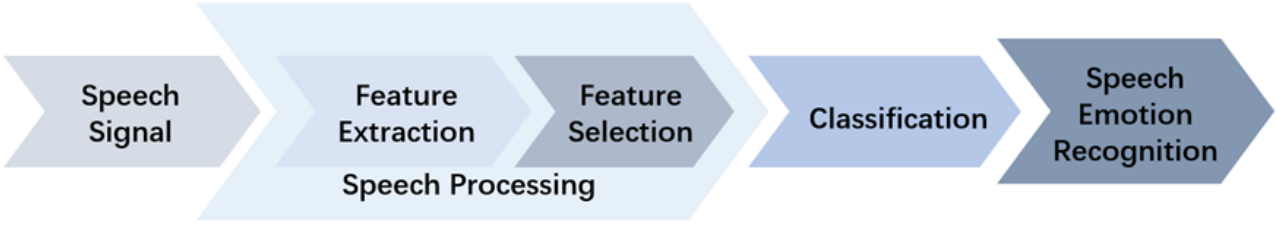

Figure 1. Experiment process flow

2.2. Data

This project uses the RAVDESS dataset [11]. The RAVDESS dataset is the Ryerson emotional speech database, which is formed by recording 24 people 60 times each. Data sets are identified by specially specified file names. The file name consists of a string of numbers, and different numbers represent different meanings to correspond to the content of this recording.

2.3. Sound feature extraction

Feature extraction of audio files is performed through Librosa, a python package for audio analysis [12]. In this study, we analyze audio by extracting its Mel-frequency cepstral coefficients (MFCCs), as well as chromagram, also with a mel-scaled spectrogram. See figure 2.

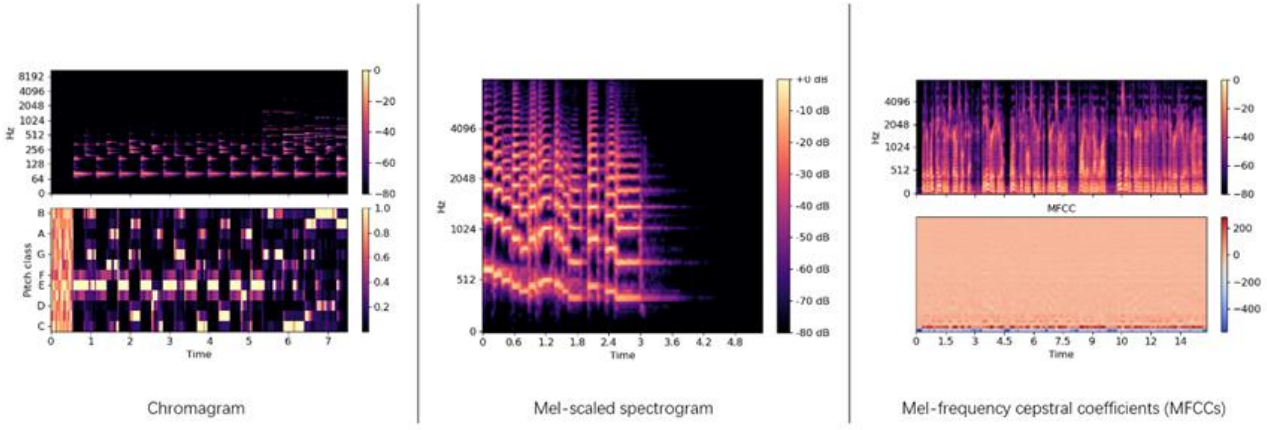

Figure 2. The Mel-frequency cepstral coefficients (MFCCs), the Chromagram [11], and the Mel-scaled spectrogram [12]

2.4. Model

This study uses an Artificial neural network (ANN) for training on the RAVDESS dataset.Multilayer Perceptron (MLP)[13] is a supervised learning algorithm, as presented as below [14]:

$$f(\cdot): R^m \rightarrow R^o \tag{1}$$

Where m is the input dimension and ois the output degree.

It is mainly used for classification or nonlinear regression. Multilayer Perceptron could have multiple non-linear layers, hidden layers, and the input and output layers. Figure 3 shows an MLP model with a single hidden layer.

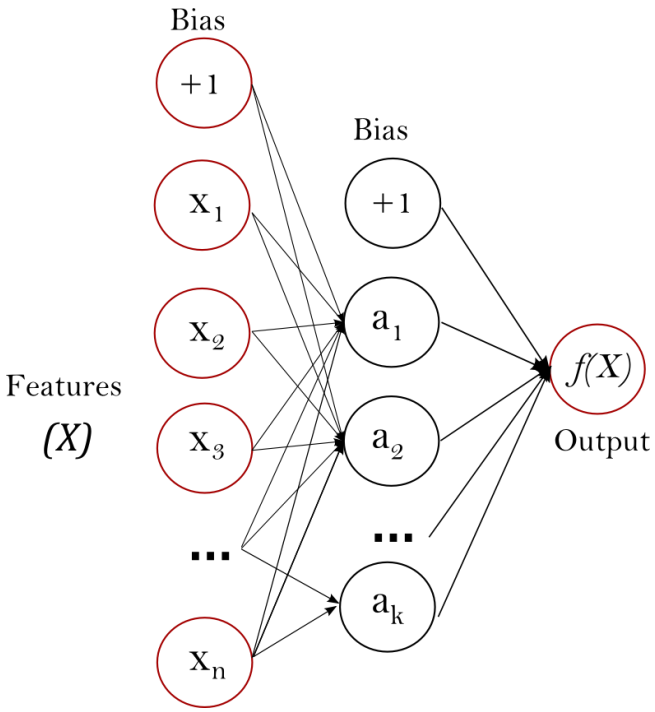

Figure 3. MLP model structure diagram with single hidden layer[14]

On the far left is the input layer, which consists of multiple neurons. Each neural unit in the hidden layer transforms the value of its previous layer using the weighted linear summation, which is then passed through a nonlinear activation function as below:

$$g(\cdot): R \to R. \tag{2}$$

Output layer accepts values from the last hidden layer then converts them. After that, the user can get the output from the output layer.

This project uses the stochastic gradient descent method (SGD), and an improved stochastic gradient descent method (SGD) 'adam' for training. It uses the gradient of the loss function with respect to the parameter to be adapted to update the parameters, as presented in Formula 3 [16]:

$$w \leftarrow w - \eta(\alpha \frac{\partial R(w)}{\partial w} + \frac{\partial Loss}{\partial w}) \tag{3}$$

ηis the learning rate, and Loss is the loss function for the network.

Based on SGD, Adam could automatically make adjustments of the amount of update parameters by adaptive estimates of the lower-order moments.

2.5. Model parameters

The optimization method can have a huge impact on the efficiency, as well as the accuracy of the model. In the early stage, this project first selects three optimization methods: ‘lbfgs’, ‘sgd’ and ‘adam’. ‘lbfgs’ is an optimizer that is in the family of quasi-Newton methods. And ‘sgd’ refers to stochastic gradient descent. Lastly, the ‘adam’ refers to a stochastic gradient-based optimizer optimized by Kingma, Diederik, and Jimmy Ba.

In order to obtain a more accurate model, this project mainly conducts experimental analysis and optimization on the following model parameters: alpha, batch size, hidden layer sizes, and max iterates.

Among them, Alpha stands for the strength of the L2 regularization, divided by the sample size when added to the loss. Batch size represents the stochastic gradient descent optimizer minimum batch size. Represents the amount of data used each time. The hidden layer sizes are the number of neurons in the hidden layer (Since this project is based on a small dataset, only a single hidden layer is used). And max iterates control the maximum number of iterations for the solver.

2.6. Experimental method

This project finds model parameters and optimization methods suitable for this dataset through the following methods:

(1)First, conduct experiments on optimization methods to find the similarities and differences of different optimization methods. Then, on the basis of determining the optimization method, the specific parameters are adjusted and tested in a wide range one by one. Finally, we try to analyze the reasons for the influence of different parameters and select the theoretically optimal parameters to generate the final model.

(2)After each parameter or optimization method is changed, the code will automatically run 10 times independently, and the running time and accuracy of each time will be calculated independently. Finally, output the average accuracy and total time. Through multiple experiments, the results are more stable and data availability is improved.

(3)Test results from comparison and selection. The result with the highest accuracy is considered the best result within an acceptable training time after multiple training sessions. If the time cost of training the model is too high, and the training accuracy is improved limited, it will not regard as the best result.

3. Discussion

3.1. Optimization method

Experiments were carried out by the method proposed in 2.6 Experimental methods, and the following results in Figure 4 were obtained for different optimization methods.

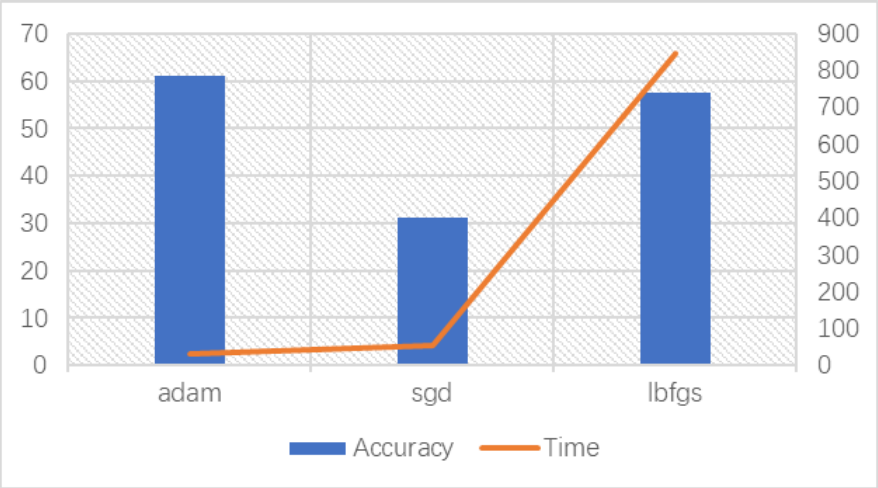

Figure 4. Accuracy and training time of model training under different optimization methods

The ‘adam’ method shows the best efficiency and accuracy for this experimental model. Under the same parameters, ‘adam’ can train a model with an accuracy of 61.17% in only 29.9 seconds. Although the ‘sgd’ method takes a short time (55.4 seconds), the model accuracy drops significantly (31.08%). At the same time, the ‘lbfgs’ method can achieve an accuracy of 57.66%, but it is trained in 847.4 seconds, and the training time is increased by more than 2820% compared with the ‘adam’ method.

The ‘adam’ optimization method has the advantages of short time and high accuracy for this project, so it will be used as the main method for training the model in this project.

3.2. Paramete

Based on the 'adam' optimization method, experiment with each variable in 2.5 by the experimental method mentioned in 2.6. Figure 5 shows the model efficiency and accuracy with different under different parameter settings. Figure 5(a), Figure 5(b), Figure 5(c), and Figure 5(d) represent the relationship between the parameter alpha, batch size, hidden layer sizes, and max iterates and the time required to train the model and the accuracy of the model.

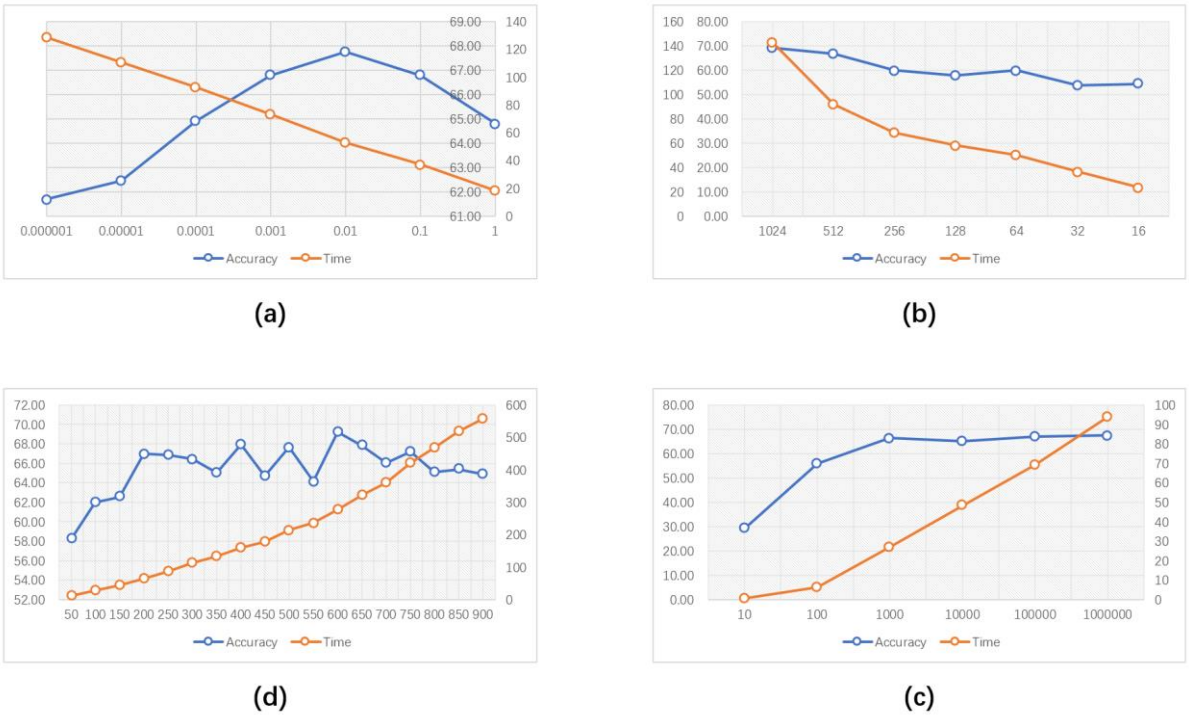

Figure 5. Efficiency and accuracy of model training under different parameters

3.2.1 Alpha

According to the experimental results shown in figure 5(a), as the alpha value becomes smaller, the time required for model training increases. The model accuracy first increases and then decreases, reaching the highest at alpha=0.01.

It can be speculated that since the alpha value affects the model complexity, the more complex the model, the more time it will take to train. Over-simplified models will lead to reduced accuracy, while over-complex models will make it difficult for the model to converge. According to the experimental data, for the data set used in this project, the complexity of the model is balanced when alpha=0.01 and the highest accuracy can be obtained without consuming too much training time.

3.2.2 Batch size

It can be seen from the experimental data in figure 5(b) that as the batch size becomes larger, the overall model accuracy shows an upward trend, and at the same time, the time required to train the model increases significantly.

When the batch size exceeds 512 to 1024, the training time is significantly increased by 50 seconds, while the model accuracy is only improved by 2.45%. Before batch size=512, although the model training time has increased significantly, it is still in a relatively efficient state, and the accuracy improvement is relatively obvious. Therefore, batch size=512 is considered to be the best parameter setting for this dataset.

The batch size represents the number of data items obtained by training the model each time. According to the test results, it can be inferred that within the test range, the smaller the amount of data obtained by each model training is, the lower the accuracy of the model results will be. More single-shot training data can improve model accuracy. At the same time, due to the increase in the amount of single training data, the time required to train the model increases accordingly.

3.2.3 Hidden layer sizes

Within the experimental range shown in figure 5(c), with the increase of hidden layer sizes, the model accuracy first increases, then oscillates, and finally decreases. The time required for model training increases linearly. From the experimental data, the accuracy of the model can reach a high level when the hidden layer sizes exceed 200. However, when hidden layer sizes reach 750, model accuracy begins to decline. After balancing the training time and model accuracy. For the dataset used in this experiment, the optimal interval for hidden layer sizes should be 200-600.

The hidden layer sizes represent how many neurons are in the hidden layer in the ANN model. From the time results, it can be inferred that for the data set of this experiment, the number of neurons below 200 is not enough to build a high-accuracy model, and the number of neurons above 750 is too complicated, which will cause the model to not converge and fail to achieve the corresponding accuracy. For training time, more neurons represent a more complex model structure, and training the corresponding model requires a larger amount of training cost, resulting in increased training time.

3.2.4 Max iterates

For Max iterates, this experiment chooses 10-1000000 as the experimental interval as shown in figure 5(d). In terms of accuracy, with the increase of max iterates, the result shows an upward trend first and then remains unchanged. In terms of time consumption, with the increase of max iterates, the result shows an upward trend. From the results, when max iterates are greater than 1000, larger values will no longer affect the model accuracy, but it will significantly increase the model training time. For the parameter max iteration, 1000-5000 is the optimal interval.

max iteration is the number of model iterations. A too low number of iterations will cause the model to stop training before reaching the best accuracy, resulting in low accuracy results. When the number of iterations is sufficient, the model reaches a steady state, that is, the best accuracy in the current situation, and more iterations will no longer affect the accuracy of the model. That is, a low number of iterations limits the model accuracy, while a sufficiently high number of iterations (1000 for this experiment) and above will no longer limit the model accuracy, and the model accuracy will

be almost independent of the number of iterations. In time, more iterations will increase the number of model training times, and at the same time increase the training time. Therefore, under the condition that the accuracy of the model is not affected, the maximum number of iterations should be as small as possible to reduce the time cost.

4. Conclusions

This project points out the current social demand for Speech Emotion Recognition (SER) and proposes an SER model. The model structure, the data structure used, and the audio analysis method are explained in detail. After that, the artificial neuron model is briefly introduced and analyzed, and the optimization methods and different parameters required in the model are proposed. After discussing the experimental method and process, the experimental results are analyzed one by one. Finally, the best model based on the dataset of this project is obtained. Based on the models and datasets used in this project, the training time and model accuracy are better balanced.

First of all, although the accuracy of 70% proves the usability of the model, there is still a lot of room for improvement in the accuracy of this model. In the future, the accuracy can be improved by increasing the sample data, improving the audio analysis method, and introducing more parameters. Also, support for different languages is worth considering.

Secondly, MLP is efficient for this dataset, however, it is not very performant and efficient all the time. During the experiment, some parameters will cause the training time to be too long, which requires a trade-off between training time and accuracy, and has to give up to further improve the accuracy of the model. The current inability to use GPUs for this model training also increases the time cost significantly when training large models. This model will need to be further trained using a more optimized algorithm in the future.

Finally, at present, this project is fully operational, and successfully analyzes the speaker's emotions through speech. However, it is not convenient for users without programming experience. Refining the project in the future and developing the UI to increase ease of use is the key to making it widely available. At the same time, packaging code for cross-platform use by other projects is also an important development direction

References

[1] Data Flair (2022). Python Mini Project. https://data-flair.training/blogs/python-mini-project-speech-emotion-recognition/. [Accessed: 29- Jul- 2022].

[2] Y. Ma, H. Drewes and A. Butz, (2022) "How Should Voice Assistants Deal With Users' Emotions?", arXiv.org. https://arxiv.org/abs/2204.02212. [Accessed: 04- Aug- 2022].

[3] Who. int, (2022) Mental disorders. https://www.who.int/news-room/fact-sheets/detail/mental-disorders. [Accessed: 04- Aug- 2022].

[4] Who.int (2022)"Mental Health ATLAS 2020", Who.int. https://www.who.int/publications/i/item/9789240036703. [Accessed: 04- Aug- 2022].

[5] Nationalcrimeagency.gov.uk, (2022). https://www.nationalcrimeagency.gov.uk/who-we-are/publications/113-prevention-and-coping-strategies-kidnapping-hostage-taking-extortion-attacks. [Accessed: 04- Aug- 2022].

[6] Hostage US. (2022). New Kidnapping Trends on the Global Stage - Hostage US. <https://hostageus.org/new-kidnapping-trends-on-the-global-stage/> [Accessed 11 August 2022].

[7] Nationalcrimeagency.gov.uk, (2022) Kidnap and extortion. https://www.nationalcrimeagency.gov.uk/what-we-do/crime-threats/kidnap-and-extortion. [Accessed: 04- Aug- 2022].

[8] B. W. Schuller, (2018) ‘‘Speech emotion recognition: Two decades in a nutshell, benchmarks, and ongoing trends,’’ Commun. ACM, vol. 61, no. 5, 90–99.

[9] M. Chen, P. Zhou, and G. Fortino, (2016) ‘‘Emotion communication system,’’ IEEE Access, vol. 5, 326–337.

[10] Livingstone SR, Russo FA (2018) The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English. PLoS ONE 13(5): e0196391. https://doi.org/10.1371/journal.pone.0196391.

[11] Ellis, Daniel P.W. (2007) Chroma feature analysis and synthesis http://labrosa.ee.columbia.edu/matlab/chroma-ansyn/

[12] McFee, Brian, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. (2015) “librosa: Audio and music signal analysis in python.” In Proceedings of the 14th python in science conference, 18-25.

[13] Scikit-learn (2011) Machine Learning in Python, Pedregosa et al., JMLR 12, 2825-2830.

[14] scikit-learn. (2022). 1.17. Neural network models (supervised). <https://scikit-learn.org/stable/modules/neural_networks_supervised.html> [Accessed 11 August 2022].