Instrument Classification Using Different Machine Learning and Deep Learning Methods

Yuqing Su *

Department of LAS, University of Illinois Urbana-Champaign, Illinois, US

* Corresponding Author Email: yuqings3@illinois.edu

Abstract. Instruments are categorized into the 5 groups in the Sachs-Hornbostel system: idiophones, membranophones, aerophones, chordophones, and electrophones. It might be easy to tell the Sachs-Hornbostel group that an instrument belongs to. However, distinguishing single instrument sound can be hard in monophonic or polyphonic music pieces and it is an important subject for musicians. Using computer science models can help musicians to analyze songs easily and fasten the speed of finding the instrument that are wanted by music producers or composers. This work aims to compare different models on particular instruments (monophonic sound) recognition which is an important problem in the field of music information retrieval. Jupyter Notebook is included for easy reproducibility. Among the six models chosen in this research: k-nearest neighbors(kNN), Support Vector Machines(SVM), Gaussian Mixture Modeling(GMM), Artificial Neural Networks(ANN), Convolutional Neural Networks(CNN) and Recurrent Neural Networks(RNN), CNN is the most accurate model and SVM is the fastest model while CNN has the prospect of being improved because it can be adjusted manually.

Keywords: Musical instrument classification, MFCCs, kNN, GMM, SVM, ANN, CNN, RNN

1. Introduction

Music is composed of different sounds which are determined by timbre. Hu- mans can easily identify some instruments with characteristic timbre or instruments that they are familiar with. However, telling apart instruments with similar timbres and recognizing instruments that are not known to the audience can be big problems. Thus, artificial intelligence comes to help and the result is beneficial for musical analysis and music composition. In 1999, 8 instruments are identified by Gaussian Mixture Models (GMM) and Support Vector Machine (SVM) with error rates 37% of GMM and 30% of SVM [1]. Further steps on instrument recognition focus on neural networks, especially Convolutional neural networks (CNN). In 2019, Solanki et el achieved 92.80% accuracy on polyphonic musical instrument recognition using CNNs with eight layers [2]. Since neural networks have developed a lot this years, more machine learning models and deep learning models were implemented and will be used in this research for comparison.

Particularly, kNN, SVM, GMM, ANNs, CNNs, and RNNs will be applied to the same instrument dataset. The instrument sound samples are from Philharmonia [3] and 7 instruments' samples are chosen to perform the research. Accuracy and time cost during model training are evaluated and analyzed. CNN is the model that will be mostly concentrated and improved. The result not only benefits the musicians, but also gives a brief look into the popular Artificial Intelligence models for computer scientists. Moreover, an assumption of sound recognition being similar to image classification in machine learning and deep learning is established after taking a look into the disadvantages and advantages of different neural networks.

In the latter parts of this paper, section 2 will discuss about data preprocessing in detail and section 3 will discuss about the six models in detail. Section 4 is the final result of the experiment and section 5 is the conclusion drawn from the results.

2. Dataset and data-preprocessing

2.1. Dataset

The dataset contains sound samples of 20 instruments [3].7instrumentsound samples are selected for convenience. The seven instruments are: guitar, flute, violin, cello, clarinet, trumpet and saxophone. The numbers of sound samples of each instrument are shown in Table 1 and unfortunately, they are not evenly distributed.

Table 1. Dataset overview

Instrument	Number of samples
guitar	106
flute	884
violin	1502
clarinet	846
trumpet	485
cello	889
saxophone	732

Samples of each instrument are separated by the rate of 7:3, which is 70% for training and 30% for testing.

2.2. Data-preprocessing

Different instruments generate sounds with different timbres. For example, the harmonicpatternoffluteisrelativelysimplewhileoboe'sharmonicpattern is complex [4]. There are many ways of extracting sound features, such as Spectral Centroid, Time Domain Zero Crossings, Mel-Frequency Cepstral Coefficients (MFCC). As a widely used method in genre classification systems, MFCC is typically believed to encode timbral information among the features it represents [5]. Besides, MFCC behaves better than the other models in Liu, Jing, and Lingyun Xie's research [6], and MFCC is chosen as the sound feature extracting method in this research. MFCC is based on the short time Fourier transform. It is widely used in various occasion of audio signal processing [6] such as music genre classification, background identification and also, instrument recognition.

The process of MFCC is presented as below:

• First, Fourier Transform is applied on the signal in the time domain of the original waveform file to get the spectrum.

• Then, logarithm is applied to the amplitude to get log-amplitude spectrum.

• Next, mel-scaling (mel filter with triangles) is used on the log-amplitude spectrum.

• Finally, discrete cosine transform, as a simplified version of Fourier Transform which can decorrelate energy in different mel-bands, is used on the result of the former step to get MFCCs features [7].

The process is visualized as below in Figure 1.

Figure 1. MFCC process

Since 12 to 20 cepstral coefficients are typically optimal for speech analysis, 13 features are extracted from MFCCs after processing the original waveform files. Each row stands for a feature in Figure 2. After the MFCC process, the sound features inside the waveform files are encoded in numbers.

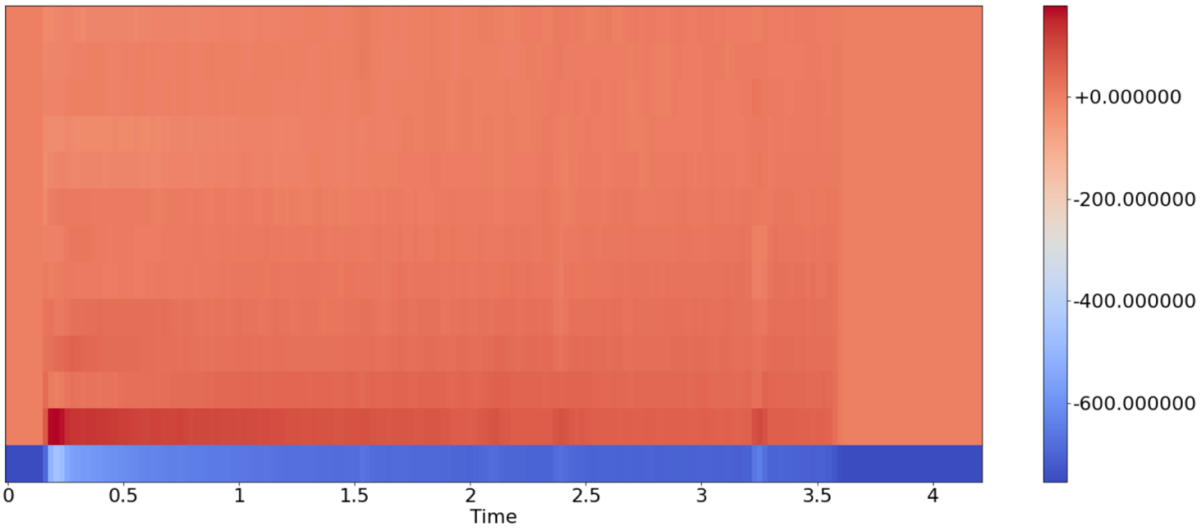

Figure 2. MFCC visualization of one waveform file

3. Classification Methods

In this study, six different classification methods are applied to fulfill the goal of instrument classification. The six methods include k-nearest neighbors(kNN), support vector machine (SVM), Gaussian Mixture Modeling (GMM), Artificial Neural Networks (ANN), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN).

3.1. kNN

kNN is one of the best known supervised learning algorithms which hinges on the hypothesis that similar objects (in this case, the sound samples of different instruments) exist in close proximity[8] and it can detect linear or non-linear distributed data. The most important value k in kNN is the number of nearest data points of a particular data point to decide which cluster that the data point belongs to and it has to be an integer. Distances between point pairs are count to ensure closer points are gathered together.

If k is too small, noise will have higher impact on the final result; if k is too large, the computation will be expensive. In this study, k is set to 4 since it reaches the highest accuracy than other number between 1 to 10. The ‘weights’ value is set to ‘uniform’ assuming that all points in each neighborhood are weighted equally.

3.2. SVM

SVM usually yields high accuracy and with relatively high speed when it comes to audio classification and segmentation [9]. It finds the optimal linear hyperplane among different dataset classes and meanwhile achieve minimum error and maximum margin.

Since there are more than two classes, the decision_function_shape parameter inside the SVM method in python needs to be considered. The decision_function_shape parameter is set to 'ovo', since 'ovo' and 'ovr' generate the same result in this research, while 'ovo' is faster. The 'ovo', One-vs-One strategy, splits a multi-class classification into one binary classification problem per each pair of classes thus there are 21 binary classification problems using SVM.

3.3. GMM

GMM is a probabilistic model that assumes all the data are generated from a mixture of a finite number of Gaussian distributions (Gaussian mixture) with the unknown parameters. The Expectation Maximization algorithm is used to adjust the parameters inside the GMM model to get the best values

of these parameters, and the process won't be discussed in detail. It is believed that GMM can help to improve the accuracy on classification problems, which have clusters of different shapes [10].

In this research, each instrument has its own GMM model. Moreover, although two GMM models for each instrument might improve the accuracy, it fails to improve the accuracy significantly [5]. Thus, only one GMM model is applied on each instrument.

3.4. ANN

ANN is a machine learning method inspired by biological neural network which uses layers of artificial neurons to transmit signals [11]. Normally, three different types of layers will exist in an ANN: input layer, hidden layers, and output layer. ANN is widely used in facial recognition and computer vision fields.

In this study, three hidden layers with rectified linear activation function (ReLU) are applied to determine the output of each hidden layer and the output layer applies softmax as the activation function. After each activation function, a dropout layer is used to avoid overfitting by randomly ignoring half of the neurons. Both the first layer and the third layer have 100 units (also are known as artificial neurons) and the second layer has 200 units. Unit number of each layer and the number of layers can be adjusted for model improvement. Number of epochs is set to 100 and batch size is set to 32 during training.

3.5. CNN

CNN is a deep learning method which mainly uses filters like small size matrices to extract features from large datasets. It is said that a ConvNet of CNN has the ability to capture the spatial and temporal dependencies comparing to other models [12]. CNN can complete tasks such as facial recognition, text digitization and natural language processing, which is assumed to be very powerful.

In this study, 4 1D-convolutional layers are applied with ReLU as the activation functions and 2 fully-connected layers are applied with ReLU as the activation function, the output layer applies softmax as the activation function. Between convolutional layers and fully-connected layers, a dropout layer is applied to avoid overfitting though this layer might be placed before the output layer. The important max-pooling matrix size is set to 2. Number of epochs is set to 100 and batch size is set to 32 during training.

3.6. RNN

RNN is a complex model which not only saves the output of processing nodes but also feeds the results back into the model [13]. The system of RNN self-learns during this backpropagation process. Thus, one of the most outstanding features of RNN is its Long Short Term Memory (LSTM). RNN is useful when it comes to text-to-speech conversions. However, it is widely known that RNN may take long time to train.

In this study, 2 LSTM layers are used followed by a dropout layer to prevent overfitting. Then 4 time-distributed dense layers are used with ReLU as the activation function. The output layer has a softmax function as the activation function. Just as in ANN and CNN, number of epochs is set to 100 and batch size is set to 32 during training.

3.7. Other methods

Apart from the six main methods that are applied in this study, Decision Tree and Decision Tree with Adaptive Boosting are also considered due to their important roles in machine learning. However, they are not the main focus of instrument recognition fields. A Decision Tree is basically a flow-chart to test on attributes of the data. And a Decision Tree can have three types of nodes: decision nodes, chance nodes and end nodes. The expected values are calculated after walking through the tree. In this research, the expected values are the predicted instrument types. The Adaptive Boosting algorithm assigns higher weights to data points which are wrongly classified to reach higher accuracy based on the original Decision Tree. And indeed, the accuracy increases from 54.00% to 74.49%.

Since these two methods are not the main focus of instrument recognition fields, they are not discussed in the result section (Section 4). Other algorithms such as Random Forests Algorithm can also be attempted.

4. Result

The overall accuracy and time spent are shown in Table 2.

Table 2. Accuracy & Time Spent Overview

Model	Accuracy	Training Time in min:sec
kNN	93.34%	00:00.01
SVM	73.35%	00:00.53
GMM	64.85%	00:33.43
ANN	88.33%	00:22.49
CNN	96.82%	00:43.57
RNN	94.83%	03:24.54

kNN is the fastest one among the six methods. It unexpectedly outperforms SVM and GMM since most studies find that SVM is normally better than kNN and GMM. One improvement is attempted by changing the 'weights' value from 'uniform' into 'distance' and the accuracy increases into 92.67%. Since 'distance' suggests that the closer neighbors of a point have greater influence than further away neighbors, it is conjectured that the sound features of instruments have some sort of inner relationships. Due to the fact the value of k and method of calculating distances can be more carefully tuned, kNN is a quite optimal method in this research. Other ways of improving kNN can be removing the outliers and normalizing the data.

SVM does a relatively good job and the time used is less than most of the other models. Inaccuracy occurs when trying to distinguish cello and violin (with similar timbre/music tone). Computer scientists usually move to SVM if kNN has low accuracy, but apparently this action shouldn't be taken in this study. SVM is believed to behave well under the condition of few points with a high dimensional space while kNN is believed to behave well under the condition of more points in a low dimensional space. The result supports this idea.

GMM, which is often used in speech recognition and was assumed to be relatively accurate, doesn't perform well especially for saxophone which only has an accuracy of 36.82%. The saxophone model recognizes the test saxophone samples evenly into other instruments. Since GMM relies highly on sample size, it might be improved if the dataset gets larger. However, due to the long time taken during models training, GMM is not a good choice for instrument recognition.

Neural networks generally outperform GMM and SVM, but require the longer running time.

ANN is the fastest neural network model in this study and generates relatively good result. It takes half of the time that CNN does.

CNN might be a better choice on instrument classification. The confusion matrix in Figure 3 shows that CNN does a good job except for trying to distinguish trumpet and saxophone which are both aerophones and can generate sounds with similar timbre by vibrating air columns. The loss-time and accuracy-time graphs in Figure 4 show that the training process gets steady around the 20th epoch.

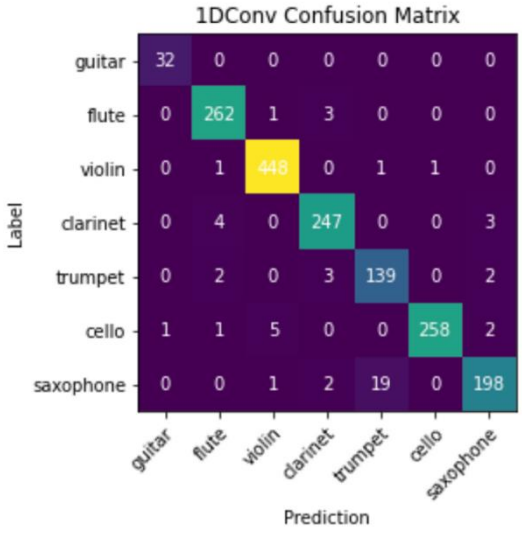

Figure 3. Confusion matrix of 1d_CNN

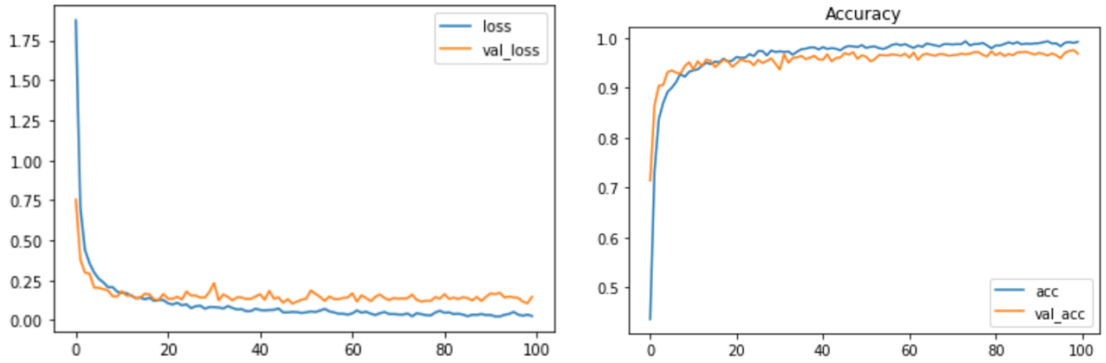

Figure 4. Loss & Accuracy over time

There are several ways to improve the accuracy. Two ways are practiced as below:

• More features are added from MFCCs results. The accuracy reaches up to 98.66% after extracting 40 features rather than 13.

• Layers are changed into 2D rather than 1D inside the model. The accuracy reaches to 94%~99% after changing 1D CNN to 2D CNN by adding the derivatives of the original MFCC result to the input layer. However, this second method does not work as the expected and is not recommended since the accuracy is unstable and the improvement is not remarkable.

RNN also achieves accurate estimation but the time cost is too high. Besides, it is not as accurate as CNN did.

5. Conclusion

Although kNN is the fastest model in this study, CNN is the most accurate one. Besides, CNN has the capability of being improved since the layers and the values inside the layers are set artificially. Also, other values such as the number of batch size can also be changed.

Moreover, ANN is said to have advantage dealing with tabular data and text data, CNN is said to have advantage dealing with image data, and RNN is said to have advantage dealing with sequence data.Since CNN clearly outperforms other models in this study, a further look into the advantages and disadvantages is taken and CNN has the superiority of recognizing the spatial relationships in the data. Thus, the result of sound data pre-processing using MFCC might share something in common with image data such as the importance of spatial features. According to this assumption, instrument classification problems and other sound recognition problems, such as background identification, music genre identification and tone recognition, can also develop as computer vision. Hence, this

study will continue by adjusting the existing models and by applying new models in other computer science fields. Another important thing to notice is that only monophonic sound samples are used in this study, the accuracy of each model can be changed if the samples are polyphonic. Recognizing the instruments in polyphonic music pieces is a further step of this study.

References

[1] Marques, J., & Moreno, P. J. (1999). A study of musical instrument classification using gaussian mixture models and support vector machines. *Cambridge Research Laboratory Technical Report Series CRL*, *4*, 143.

[2] Solanki, A., & Pandey, S. (2019). Music instrument recognition using deep convolutional neural networks. *International Journal of Information Technology*, 1-10.

[3] Philharmonia Limited Registered Charity. 2022. *Sound samples* [dataset]. https://philharmonia.co.uk/resources/sound-samples/

[4] Bernier, J. J., & Stafford, R. E. (1972). The relationship of musical instrument preference to timbre discrimination. *Journal of Research in Music Education*, *20*(2), 283-285.

[5] Solanki, A., & Pandey, S. (2019). Music instrument recognition using deep convolutional neural networks. *International Journal of Information Technology*, 1-10.

[6] Liu, J., & Xie, L. (2010, May). SVM-based automatic classification of musical instruments. In *2010 International Conference on Intelligent Computation Technology and Automation* (Vol. 3, pp. 669-673). IEEE.

[7] Li, T. L., & Chan, A. B. (2011, January). Genre classification and the invariance of MFCC features to key and tempo. In *International Conference on MultiMedia Modeling* (pp. 317-327). Springer, Berlin, Heidelberg.

[8] Harrison, O. 2019. Machine learning basics with the K-nearest neighbors algorithm. https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761

[9] Lu, L., Zhang, H. J., & Li, S. Z. (2003). Content-based audio classification and segmentation by using support vector machines. *Multimedia systems*, *8*(6), 482-492.

[10] Kumar, A. 2022. Gaussian mixture models: What are they & when to use? https://vitalflux.com/gaussian-mixture-models-what-are-they-when-to-use/#:~:text=Gaussian%20mixture%20models%20(GMMs)%20are,marketing%20and%20so%20much%20more!

[11] Shreevathsa, P. K., Harshith, M., & Rao, A. (2020, January). Music instrument recognition using machine learning algorithms. In *2020 International Conference on Computation, Automation and Knowledge Management (ICCAKM)* (pp. 161-166). IEEE.

[12] Saha, S. 2018. A comprehensive guide to Convolutional Neural Networks - the eli5 way. https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

[13] Gupta, A. 2022. Difference between ANN, CNN and RNN. https://www.geeksforgeeks.org/difference-between-ann-cnn-and-rnn/#:~:text=ANN%20is%20considered%20to%20be,compatibility%20when%20compared%20to%20CNN.&text=Facial%20recognition%20and%20Computer%20vision