Chun Pui Tang, Ka Long Chui, Ying Kin Yu, Zhiliang Zeng, Kin Hong Wong, "Music Genre classification using a hierarchical Long Short Term Memory (LSTM) model", International Workshop on Pattern Recognition IWPR 2018 ,University of Jinan, Jinan, China, May 26-28, 2018

Music genre classification using a hierarchical long short term memory (LSTM) model

Chun Pui Tang, Ka Long Chui, Ying Kin Yu, Zhiliang Zeng, and Kin Hong Wong

Department of Computer Science and Engineering, The Chinese University of Hong Kong

ABSTRACT

This paper examines the application of Long Short Term Memory (LSTM) model in music genre classification. We explore two different approaches in the paper. (1) In the first method, we use one single LSTM to directly classify 6 different genres of music. The method is implemented and the results are shown and discussed. (2) The first approach is only good for 6 or less genres. So in the second approach, we adopt a hierarchical divide-and-conquer strategy to achieve 10 genres classification. In this approach, music is classified into strong and mild genre classes. Strong genre includes hiphop, metal, pop, rock and reggae because usually they have heavier and stronger beats. The mild class includes jazz, disco, country, classic and blues because they tend to be softer musically. We further divide the sub-classes into sub-subclasses to help with the classification. Firstly, we classify an input piece into strong or mild class. Then for each subclass, we further classify them until one of the ten final classes is identified. For the implementation, each subclass classification module is implemented using a LSTM. Our hierarchical divide-and-conquer idea is built and tested. The average classification accuracy of this approach for 10-genre classification is 50.00%, which is higher than the state-of-the-art approach that uses a single convolutional neural network. From our experimental results, we show that this hierarchical scheme improves the classification accuracy significantly.

Keywords: Computer Music, LSTM, Music Genre Classification

1. INTRODUCTION

Nowadays, machine learning has been widely applied to many different fields, for examples healthcare, marketing, security and information retrieval. Artificial neural network is one of the most effective techniques that is good at solving classification and prediction problems. In this project, we apply an artificial neural network to music genre classification. Our target is to classify music into different genres, for example 6-10 different genres. Our algorithm is very useful for the user to search for their favorite music pieces. The applications of machine learning techniques to music classification is not common compared to image classification. Tao et. al.[1] created a deep learning model that can identify the music from at most 4 different genres in a dataset. The method is also mentioned in a journal paper.[2] In this project, we make use of the Long Short-Term Memory (LSTM) model instead of CNN in the music genre classification problem. We are able to train a model that can classify music from 6 to 8 different genres. Furthermore, we adopt a divide-and-conquer scheme to further improve the accuracy. Firstly, we divide the music into two classes, namely the strong and mild class. A LSTM classifier is trained to categorize music into these two classes. Then the music is further classified into a number of subclasses. From our experimental results, we show that this hierarchical scheme improves the classification accuracy.

Our paper is organized as follows. In Section 2, we introduce the background of our work. In Section 3, the theory used is discussed. In Section 4, we describe the implementation details and the classification results of our LSTM approaches. The discussion and conclusion are found in Sections 5 and 6, respectively.

2. BACKGROUND

Research related to music is interesting and has many commercial applications. Machine learning can provide elegant solutions to some problems in music signal processing, such as beat detection, music emotion recognition and chord recognition. In 2013, Van den Oord etal[3] published a survey on deep learning in music. Moreover,

*E-mail: khwong@cse.cuhk.edu.hk, This work is supported by a direct grant (Project Code: 4055045) from the Faculty of Engineering of the Chinese University of Hong Kong.

Table 1. Results of the system described in[1]

Number of Genres	Testing set
2 genres (Classic, Metal)	98.15%
3 genres (Classic, Metal, Blues)	69.16%
4 genres (Classic, Metal, Blues, Disco)	51.88%

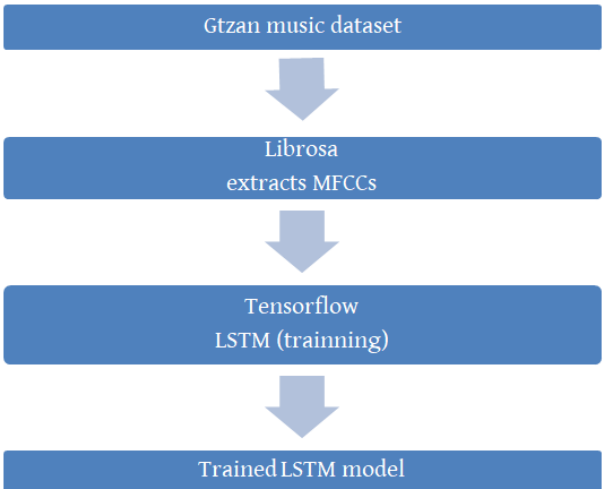

Figure 1. An overview of the music genre classification process.

Arora et al[4] developed a project on environmental sound classification. Also, a music composition robot called BachBot is created by Feynman Liang.[5] It uses LSTM to create J.S. Bach style music pieces.

In this paper, we are interested in applying machine learning to music genre classification. Feng et. al.[1] devised an algorithm to classify music into in 2 to 4 genres. Their results are summarized in Table 1. From the table, we can see that the accuracy of classifying 2 genres is 98.15%. However, when the number of genres is increased to 4, the accuracy is reduced by 17%. Another work that tackles the same problem is proposed by Matan Lachmish.[6] Their approach uses the convolutional neural network model. They achieved an accuracy of 46.87% in classifying music into 10 different genres. In this paper, we are going to use the LSTM model to solve the music classification problem. To the best of our knowledge, we believe that we are one of the first groups to solve this problem using LSTM.

3. THEORY

Figure 1 illustrates the overall structure of our project. We use the Gtzan[7] music dataset to train our system. We apply the Librosa library[8] to extract audio features, i.e. the Mel-frequency cepstral coefficients (MFCC), from the raw data. The extracted features are input to the Long Short-Term Memory (LSTM) neural network model for training. Our LSTM are built with Keras[9] and Tensorflow.[10]

3.1 Mel frequency cepstral coefficients (MFCC)

MFCC features are commonly used for speech recognition, music genre classification and audio signal similarity measurement. The computation of MFCC has already been discussed in various papers.[11] We will focus on how to apply the MFCC data for our application. In practice, we use the Librosa library to extract the MFCCs from the audio tracks.

3.2 The Long Short Term Memory Network (LSTM)

Approaches such as Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) are popular machine learning framework. The LSTM network used in this project is a subclass of RNN. RNN is different from the traditional neural networks. It can memorize the past data and is able to predict with the help of the information stored in the memory. Moreover, LSTM solves the RNN long term dependencies problem. Although

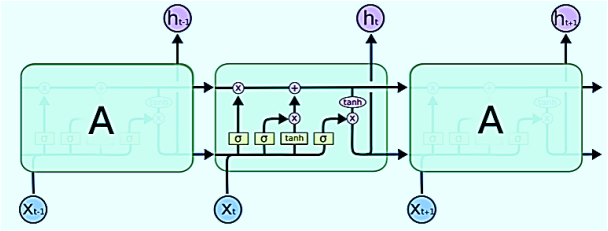

Figure 2. A typical LSTM model contains four interacting layers.[12]

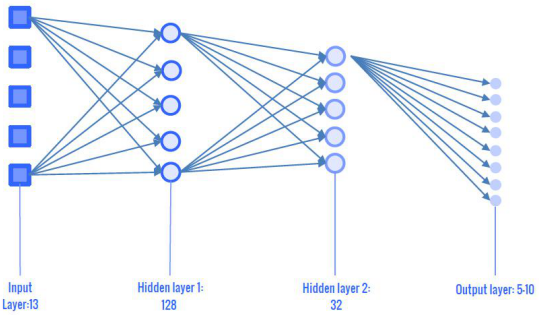

Figure 3. The LSTM network used in our music genre classification problem.

RNN model can make use of the past information to predict the current state, the RNN model may fail to link up the information when the gap between the past information and the current state is too large. The details of the long-term dependencies have been discussed in a tutorial,[12] Figure 2 reveals the structure of a typical LSTM model. Figure 3 shows the configuration of our LSTM network. The network has 4 layers. A LSTM can be formulated mathematcially as follows:

$$
\begin{aligned}
&u_t = \tanh(W_{xu} * x_t + W_{hu} * h_{t-1} + b_u) : \text{update equation} \\
&i_t = \sigma(W_{xi} * x_t + W_{hi} * h_{t-1} + b_i), \text{input gate equation} \\
&f_t = \sigma(W_{xf} * x_t + W_{hf} * h_{t-1} + b_f), \text{forget gate equation} \\
&o_t = \sigma(W_{xo} * x_t + W_{ho} * h_{t-1} + b_o), \text{output gate equation} \\
&c_t = i_t \odot u_t + f_t \odot c_{t-1}, \text{cell state} \\
&h_t = \tanh c_t \odot o_t, \text{cell output} \\
&output_{class} = \sigma(h_t * W_{outpara})
\end{aligned}
\tag{1}
$$

where W_{xu}, W_{xi}, W_{xf}, W_{xo} and W_{hu}, W_{hi}, W_{hf}, W_{ho}, $W_{outpara}$ are weights, and b_u, b_i, b_f, b_o are biases to be computed during training. h_t is the output of a neuron at time t. $\odot$ denotes pointwise multiplication. $\sigma()$ denotes a sigma function and tanh() represents the tanh function. The input x_t is the MFCC parameters at time t. $output_{class}$ is the classification output.

4. IMPLEMENTATION AND EXPERIMENTAL RESULTS

4.1 Our dataset

We used the GTZAN dataset[13] that contains of various samples of the ten music genres in our experiments. The genres are blues, classic, country, disco, hip-hop, jazz, metal, pop, reggae and rock. Each genre includes

Table 2. The design of our LSTM network in experiment 1.

Input Layer (I)	13 MFCCs features obtained as input
Hidden layer (II)	128 neurons
Hidden Layer (III)	32 neurons
Output Layer (IV)	6 outputs corresponding to 6 different genres of music

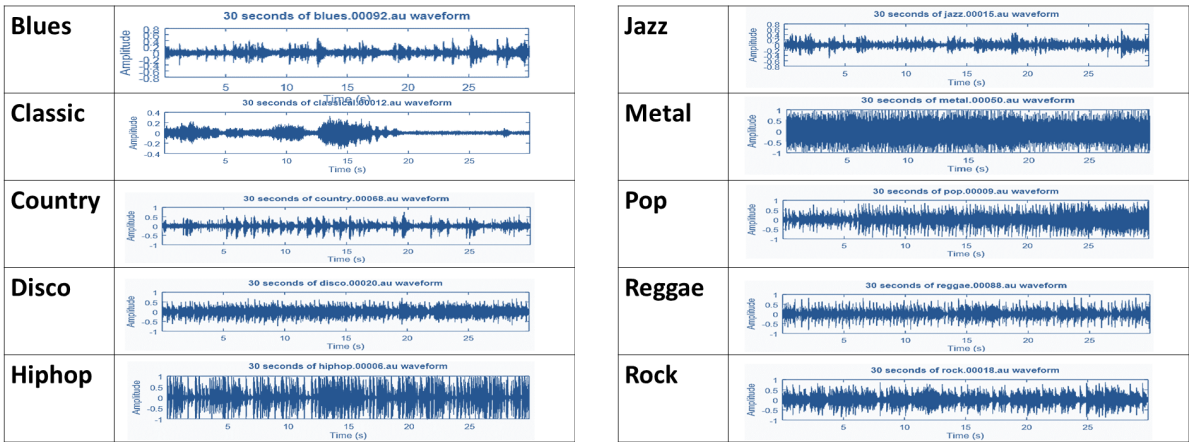

Figure 4. Sample waveforms of different music genres.

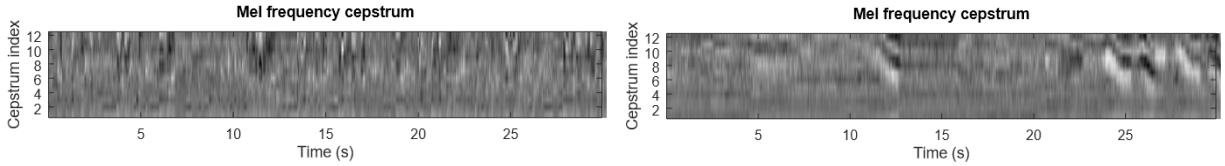

Figure 5. Visualization of Mel frequency cepstrum.

100 soundtracks of 30 seconds long in .au format. We randomly chose samples from the dataset for training and testing. Using the script written by Kamil Wojcicki,[14] we created the waveforms of the soundtracks and compared their similarity. Samples of the waveforms are shown in Figure 4. 30% of the data are used for testing and 70% of the data are used for training. The testing and training dataset are not overlapped.

We compared the waveforms of 10 different genres. It is found that blues is similar to jazz and country. Rock is similar to pop and reggae. So we decided to use music from the classic, hip-hop, jazz, metal, pop and reggae to form the six genres for training in our first experiment.

4.2 Preprocessing

Before we can use the data in the GTZAN dataset, we need to preprocess the signals so that they can be input to the Long Short Term Memory (LSMT) model. MFCC is a good representation of music signals. It is one of the best indicators of the 'brightness' of the sound. In practice, it is able to measure the timbre of the music by the method discussed in the paper by Emery Schubert etal.[15] We used the Librosa library[8] to transform the raw data from GTZAN into MFCC features. In particular, we chose the frame size as 25ms. Each 30-second soundtrack has 1293 frames and 13 MFCC features, which are C1 to C13 in experiment. There are 14 MFCC features, which are C0 to C13 in experiment 2. Figure 5 shows some examples of the Mel frequency cepstrum plots of the music signals in the database.

4.3 Experiment 1 : LSTM for 6-genre classification

In this experiment, there are 420 audio tracks in the dataset for training, 120 for validation and 60 for testing. Each audio track lasts for 30 seconds. We set the batch size that defines the number of samples to be propagated through the network for training as 35. We can see that the accuracy and loss are improving within 20 Epochs. At 20, the test accuracy reaches the maximum and the loss is minimized. We achieved a classification accuracy of around 0.5 to 0.6. There are still some rooms for improvement. With more training samples, we may be able to achieve an accuracy of 0.6 to 0.7. The major limitation is the small training data size. It leads to low accuracy and overfitting. Although some genres, such as metal, are outstanding and easy to be recognized, it is hard to classify some other genres that are quite similar.

From Figure 6, there is a overlapping of some features among different genres. For instance, we can observe

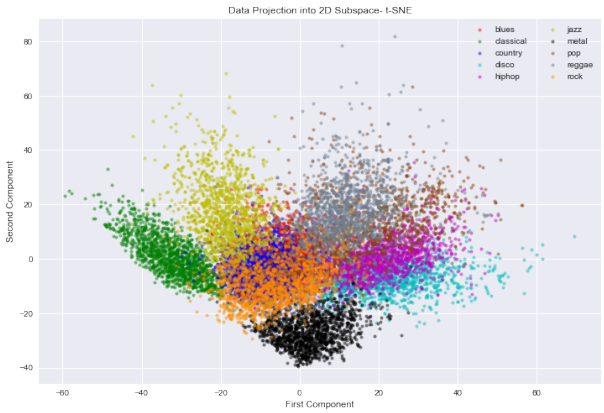

Figure 6. Classification of music genres in GTZAN dataset[16]

,

that the data points of pop music overlap with other genres. It is reasonable because pop songs include features of other genres.

4.4 Experiment 2 : The hierarchical approach for 10-genre classification

A divide-and-conquer scheme is employed. In our scheme, we applied 7 LSTMs. Then a multi-step classifier involving all these 7 LSTM classifiers were used to achieve 10-grenre classification. The division of samples for training and testing is the same as that in experiment 1. The LSTM classifiers involved are listed below.

- LSTM1: It classifies music into strong (hiphop, metal, pop, rock and reggae) and mild (jazz, disco, country, classic and blues) group.
- LSTM2a: It divides the music into Sub-strong1 (hiphop, metal and rock) and Sub-strong2 (pop and reggae) classes. During training, only music samples of hiphop, metal, rock, pop and reggae are involved.
- LSTM2b: It categorizes music into Sub-mild1(disco and country) and Sub-mild2 (jazz, classic and blues) groups. We used samples only from disco, country, jazz, classic and blues for training.
- LSTM3a: It classifies music into hiphop, metal and rock. Only music from hiphop, metal and rock class are involved.
- LSTM3b: It differentiates pop music from reggae
- LSTM3c: It differentiates disco music from country.
- LSTM3d: It recognizes jazz, classic and blues.

The proposed multi-step classifier involves the 7 LSTMs above. In the testing stage, the input music is first classified by LSTM1 to find if it is strong or mild. Then according to the result, either LSTM2a or LSTM2b is applied. Finally, LSTM3a, 3b, 3c or 3d is used to classify the music into the target categories according to the results obtained in the previous level. Results of this experiment are shown in Table 3. Our approach achieved an accuracy of 50.00%. It was better than the state-of-the-art approach based on convolutional neural network, which had an accuracy of 46.87%.[6] A diagram showing the hierarchy of the LSTMs in our multi-step classifier is shown in Figure 7.

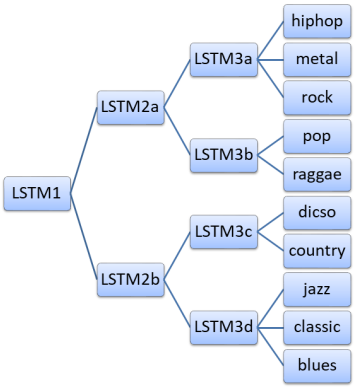

Figure 7. The hierarchy of the LSTMs in our multi-step classifier.

,

Table 3. Results of experiment 2. The accuracy of each LSTM component in the proposed multi-step classifier.

LSTM classifiers	Accuracy	Epochs
LSTM1 (strong, mild)	80.0%	35
LSTM2a (sub-stong1, sub-strong2)	81.6%	20
LSTM2b (sub-mild1, sub-mild2)	81.6%	35
LSTM3a (hiphop, metal, rock)	74.6%	40
LSTM3b (pop, reggae)	88.0%	20
LSTM3c (disco, country)	78.0%	20
LSTM3d (jazz, classic, blues)	84.0%	40
Our multi-step classifier for all 10 genres	50.0%	N/A

5. CONCLUSION

In conclusion, the experimental results show that our multi-step classifier based on Long Short-Term Memory (LSTM) model is effective in recognizing music genres. For 6-genre classification, the accuracy was 50-60% using a single LSTM. We also used a divide-and-conquer approach to classify 10 genres of music. We achieved an accuracy of 50.00%, which was better than one of the state-of-the-art approaches having an accuracy of 46.87%.[6]

REFERENCES

[1] Tao Feng. Deep learning for music genre classification, university of illinois.[online]. https://courses.engr.illinois.edu/ece544na/fa2014/Tao_Feng.pdf. Accessed: 16- April- 2018.

[2] Lin Feng, Sheng-lan Liu, and Jianing Yao. Music genre classification with paralleling recurrent convolutional neural network. *CoRR*, abs/1712.08370, 2017.

[3] Aaron van den Oord, Sander Dieleman, and Benjamin Schrauwen. Deep content-based music recommendation. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2643–2651. Curran Associates, Inc., 2013.

[4] Raman Arora and Robert A Lutfi. An efficient code for environmental sound classification. *The Journal of the Acoustical Society of America*, 126(1):7–10, 2009.

[5] Feynman Liang. *BachBot: Automatic composition in the style of Bach chorales*. PhD thesis, Masters thesis, University of Cambridge, 2016.

[6] Mlachmish. Music genre classification with cnn. https://github.com/mlachmish/MusicGenreClassification/blob/master/README.md. Accessed: 16- April- 2018.

[7] Bob L Sturm. An analysis of the gtzan music genre dataset. In *Proceedings of the second international ACM workshop on Music information retrieval with user-centered and multimodal strategies*, pages 7–12. ACM, 2012.

[8] Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, pages 18–25, 2015.
[9] Francois Chollet. *Deep learning with Python*. Manning Publications Co., 2017.
[10] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016.
[11] Md Sahidullah, Sandipan Chakroborty, and Goutam Saha. Improving performance of speaker identification system using complementary information fusion. *arXiv preprint arXiv:1105.2770*, 2011.
[12] Christopher Olah. Understanding lstm networks. *GITHUB blog, posted on August*, 27:2015, 2015.
[13] GTZAN. Gtzan genre data set. http://marsyasweb.appspot.com/download/data_sets/. Accessed: 16-April- 2018.
[14] Kamil Wojcicki. Htk mfcc matlab. *MATLAB Central File Exchange*, 2011.
[15] Emery Schubert, Joe Wolfe, and Alex Tarnopolsky. Spectral centroid and timbre in complex, multiple instrumental textures. In *Proceedings of the international conference on music perception and cognition, North Western University, Illinois*, pages 112–116. sn, 2004.
[16] Arthur Flexer. Improving visualization of high-dimensional music similarity spaces. In *ISMIR*, pages 547–553, 2015.