MLP Coursework 3: Deep Networks Music Classification Based on FMA

s1738623, sXXXXXXX, sXXXXXXX

Abstract

The abstract should be 100–200 words long, providing a concise summary of the contents of your report.

1. Introduction

Machine learning is âĂę

Deep learning(DL) is a subset of machine learning and there are three main types of learning: supervised, semi-supervised and unsupervised. In a deep learning model, there are multiple layers each of whose input is from the former layerâĂŹs output. Many of current deep learning models are based on Artificial neural network (ANN) which includes deep neural networks (DNNs). A deep neural network contains many hidden layers between the input layers and output layers. One of advantages of DNNs is the ability of finding non-linear relationships which are often complex. Besides DNN, there are some other deep learning algorithms, such as Recurrent neural networks (RNNs) and Convolutional deep neural networks (CNNs). RNNs allows data going in any direction and are mainly used as language model. CNNs are mainly used in computer vision domain.

Music information retrieval(MIR), as a branch of information retrieval(IR), is the science of retrieving information from music. There are many developed IR application in the world, for example, the text information retrieval from search engine. Unlike text information retrieval, MIR research is still not much due to many reasons. One of the reason above is MIR often requires a comprehensive background in music, psychology, machine learning etc. Another reason is the lack of numbers of large, complete and available datasets. (**?**).

In the rest of the report, there are mainly eight sections. Motivation, research questions and objectives will be stated in section 2 to 4 respectively. Section 5 is a brief introduction on FMA datasets. In methodology part (section 6), a concise conclusion of machine learning practical will be given and some novel knowledge used in our experiment will be introduced as well. After experiments part, there is a interim conclusion containing results analysis. At the last, future plan of the project is pointed out.

2. Motivation

Machine learning performs good at classification, and deep learning often has a better performance on finding non-linear relationships. Music is a kind of thing containing a lot of information which is hard to find the relative relationships inside. Therefore, compared other methods, machine learning, especially deep learning, is a suitable approach to deal with music issues.

Besides deep learning or machine learning, music, which is a natural habit of human, has a remarkable potential market to be explored. For example, a good music classification can bring clear learning schedules for new learner. Also, music recommend systems based on personal preference will be popular because of the increasing cognition of self-value.

Therefore, in this report, our motivation is to find an approach based on deep learning to classify the music well. Besides that, we also want to find some other classification models and make some comparison. All the experiments are based on FMA datasets and classification accuracy is the evaluation of models.

3. Research questions

Unlike the wealth of other information retrieval, lacking large, complete and available datasets for MIR makes MIR research develop slow. Except for FMA dataset, other existing datasets for MIR are either in small scale or not complete. For example, despite containing 2524739 clips, the dataset named AcousticBrainz has no information on artists or audios. Dataset called Unique provides a more complete information, but its capacity(3115 clips) is far away from the FMA(106547 clips). (**?**)

dataset	clips	artists	year	audio
RWC	465	–	2001	yes
CAL500	500	500	2007	yes
Ballroom	698	–	2004	yes
GTZAN	1000	300	2002	yes
MusiClef	1355	218	2012	yes
Artist20	1413	20	2007	yes
ISMIR2004	1458	–	2004	yes
Homburg	1886	1463	2005	yes
103-Artists	2445	103	2005	yes
Unique	3115	3115	2010	yes
1517-Artists	3180	1517	2008	yes
LMD	3227	–	2007	no
EBallroom	4180	–	2016	no
USPOP	8752	400	2003	no
CAL10k	10271	4597	2010	no
MagnaTagATune	25863	230	2009	yes
Codaich	26420	1941	2006	no
FMA	106574	16341	2017	yes
OMRAS2	152410	6938	2009	no
MSD	1000000	44745	2011	no
AudioSet	2084320	–	2017	no
AcousticBrainz	2524739	–	2017	no

A complete, large dataset containing more information may reveal better relationships between data by fitting them into deep networks. Therefore, our project, which is based on FMA, aims at build MIR classification models in a new, better data environment than before. Also, in order to find a suitable model, we build different models and compare them by their accuracy. Although FMA, which contains both text and audio information both, provides a good audio resource, we still want to explore the text information in this report. The reason for this is mainly about the casual habits that ordinary people often are used to classify the types of music by artists, albums and others belonging to text information. Therefore, we really want to figure out one or several models classifying the music type based on these text characters.

4. Objectives

In this interim report, we mainly illustrate the relationship among the machine learning, deep learning and music information retrieval. After a comprehensive research, we have found some problems leading to several worthy research points. As the priority, we want to build a classification based on DNNs by using the FMA dataset. Besides that, there are two optional goals as well. One is finding more suitable classification models based on the FMA dataset, such as RNNs etc. The other one is trying to build a recommending model based on our classification results.

5. Data

The dataset in our experiment is called Free Music Archive (FMA). It is a dataset opening for free and suitable for evaluating various MIR tasks. FMA dataset contains both texture and audio content inside. For texture content encode as csv format, there are some documents recording different information about songs, albums, artists etc. For audio content encoded as mp3, there are four different size of packages, which contains 8000 tracks in 30s (7.2GiB), 25000 tracks in 30s (22GiB), 106574 tracks in 30s (93GiB) and 106574 tracks untrimmed (879GiB) respectively.

In this report, as a mid-term report, our work is mainly about the classification issues. We extract the main text features, such as article, created date, length etc., from the dataset and split it into train set, validation set and test set. Then build different models to fit the data respectively. The evaluation of our task is accuracy of classification.

6. Methodology

7. Experiments

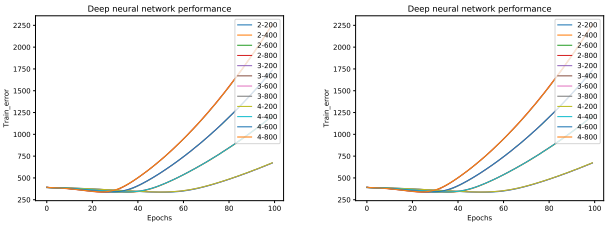
After the data pre-processing, we started to implement our baseline experiments. Our main idea is to test the data set with some normal deep neural networks (DNN) we have built during coursework 1 and 2. The reason why we choose only DNN to perform our baseline experiments is that DNN has been well tested and understood by all of us. Meanwhile, we are not familiar with the data set. If there are some unexpected errors in the results, we can focus on solving the data set problems rather than worrying about both neural network problems and data set problems.

In this section, we will present our baseline experiments in time line order. We will first introduce the parameters we used to set up the DNNs. Then, we will describe our three group of baseline experiments and what we learned from them.

7.1. Neural network structure

Our DNN structure is quite simple. We used leaky-ReLU as the activation function in the neural network. We chose the Adam optimizer as our optimizer. We also defined the loss as the "softmax cross entropy with logits". We also include the batch normalization and dropout layer in our DNN. The dropout rate is set to 0.2. The learning rate, hidden layer number and the number of neurons in the hidden layer will be varied during the experiments.

After we constructed our neural network, we tested it on the EMINST data set with learning rate is 0.2, hidden layer is 3 with each layer has 500 neurons. The results are shown as following:



(a) Error (b) Accuracy

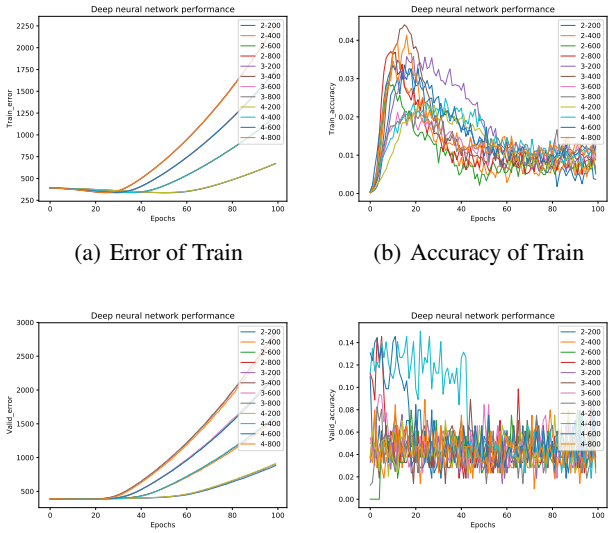
Figure 1. The EMINST on DNN

Which shows that our neural network is constructed correctly.

7.2. Experiments with full data

After the neural network testing, we then provide the FMA data set to our DNN. We varied the hidden layer numbers to be 2, 3, 4 with the number of neurons in each hidden layer to be 200, 400, 600 and 800 respectively. Each neural network will be trained in 100 epochs. Which means we have $3 \times 4 \times 100 = 1200$ epochs in total. The results are shown as following:

As the Figure.2 shown, both the error of training set and validation set are increasing, which indicates that there is something wrong and the DNN totally unable to learn the features.



(a) Error of Train (b) Accuracy of Train

(c) Error of Valid (d) Accuracy of Valid

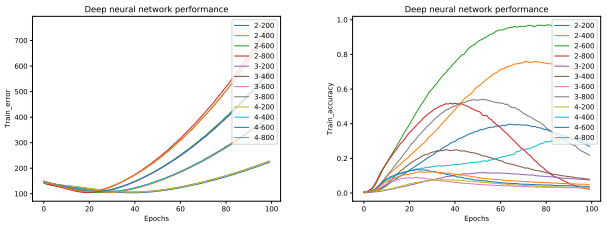
Figure 2. The error and accuracy curves of both training and validation sets with full data tested

At the beginning, we thought it might be the learning rate problem. However, after we carefully searched, the learning rate have been set to a proper value and the neural network still cannot learn. The results are unacceptable. Even if DNN is not suitable for this task, the error on training set should not increase as the increasing of epochs. Since the DNN worked well in EMINST data set, we believed there should be something wrong with the data set.

After our carefully checking, we noticed the data set is not balanced. As mentioned before, some songs belong to more than one genres. In our data pre-processing procedure, we mapped the multiple genres into new single genre so that we can implement one-hot coding. However, this increase the number of total genres vastly. The original number of genres are 236 in total. After our pre-processing, the total genre number is 4858, which is about 8 times of our input feature dimensions (which is 518 in total.). Therefore, our neural network cannot perform the classifying work. Because the output dimension is even larger than the input dimension. Therefore, we decide to decrease the number of genres.

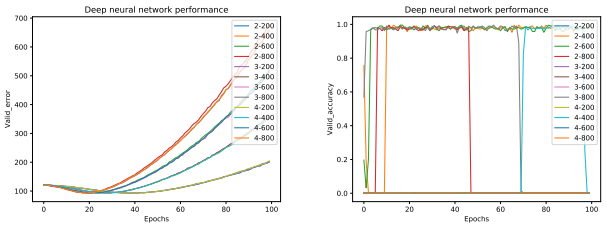
7.3. Experiments with reduced data

To decrease the number of genres without removing any data, we decide to merge the genres. If one song has more than one genres, we only left the first genre it mentioned as its genre. By doing so, we deceased the number of genres from 4858 to 236, which is smaller than the input dimension. With all parameters unchanged, we implement our experiments again on the reduced data set.The results are shown as following:



(a) Error of Train

(b) Accuracy of Train



(c) Error of Valid

(d) Accuracy of Valid

Figure 3. The error and accuracy curves of both training and validation sets with reduced data tested

As the Figure.3 shown. The results are still quiet bad. By looking at the error of training set, we can infer that there is still something wrong with the data set. Although the error of training set is decreasing at the beginning, it is also increasing at the end. We think no matter how bad the neural network is in dealing with the classifying work, the error of training set should not increase at any time.

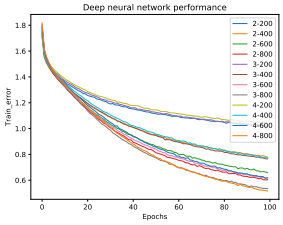
Meanwhile, the accuracy of validation set is quite interesting. Some of the situation, the accuracy of validation set is suddenly increased to about 1 (100% accuracy) and suddenly decreased to zero again. We guessing this is because when we cutting the data, there is something wrong with our shuffle process which cause the data in validation set is not shuffled. Therefore, since only one or two genres in the validation set, the accuracy rate will be look like this.

Later, we found that we forgot to shuffle the reduced validation set. Due to the time limitation, we cannot implement this experiments again. However, since we shuffle the data separately, other data sets are not affected.

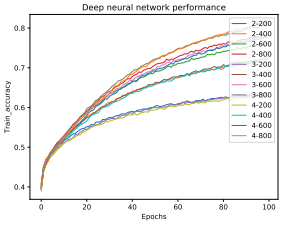
After testing with various learning rates and other parameters, we thought it is still the data set problem. Inspired by EMINST data set (which input dimension is 20 times to output dimension) and MINST data set (which input dimension is 70 times to output dimension), we decide to decrease the number of genres even more.

7.4. Experiments with deleted data

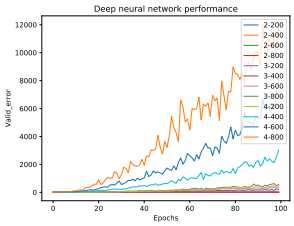
Since we have reduced the data set, it is impossible to decrease the number of genres without deleting data. Therefore, we decide to use the top 10 genres in the data set. If a song is not belong to one of the top 10 genres, we will delete it from the data set. By constructing our data set in this way, we decreased the number of genres to 10 with saving most of the data.With all parameters unchanged, we implement our experiments again on the deleted data set.The results are shown as following:



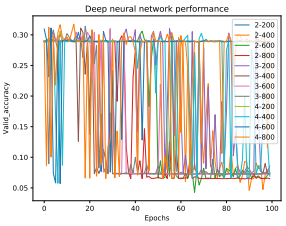
(a) Error of Train



(b) Accuracy of Train



(c) Error of Valid



(d) Accuracy of Valid

Figure 4. The error and accuracy curves of both training and validation sets with deleted data tested

As the Figure.4 shown. The results are finally normal. The error on the training set keeps decreasing and the accuracy on the training set keeps increasing. Which indicates that our neural network model is working with the data set. Which means that our guessing before was right.

However, the results also shows that the data set has another problem. As the error of the validation set keeps increasing rapidly, it is clear that our neural network overfitted to the training data very quickly. What is worse, from the accuracy of the validation set keeps shaking around rather than getting worse, it is clear that the inputs are not quite related to the outputs. In another word, the features in the metadata is not related to the genres of the songs.

In conclusion, during our baseline experiments, we found that the features provided by the FMA project has no strong relationship with the genre of the song. What is worse, the features provided by FMA also have no relationship to time, which means it is hard for us to implement convolutional neural network on this data set. Therefore, we will implement our own music feature function to obtain the features from music files directly and re-implement our experiments with other neural network such as RNN, LSTM and 1-D convolutional neural network.

7.5. #COPY THIS TO DATA PART#

(Following I provided some data pre-processing information to you guys just in case you need it. DO copy it to the data part and use it as you like. DO rephrase it if it sounds weird, I didn't write this as formal form.)

The data from FMA needs to be pre-processed. As men-

tioned before, it has four files in metadata part. They contain the data of tracks, genres, features and echonest. Since our goal in the baseline part is to verify the relationship between genres and features (we want to give the genres based on the melody of the songs), we do not need echonest file at this stage.

To construct our training data, we first fetched the genres of each song. In the track file, some of the songs may have exact one genre, others may have more than one genres or no defined genre. Therefore, we add a new genre for those songs with more than one genre. (e.g. if a there is 168 different genres in total, one song has both genre 16 and 32. We will define a new genre named 169 and change all songs with both genre 16 and 32 to genre 169.) Which means our neural network will only be considered as correct if it predict all genres right. Then, since the genres are given by numbers defined online rather than ordered numbers (which means that the genre number is not consecutive), we went through the whole data set to reorder the genres of the songs. After that, with the proper ordered numbers, we implement the normalization on the input data set, which is the feature file. Finally, we compressed the whole data set to .npz format so that it can be used by our data providers.

8. Interim conclusions

9. Plan