Music to Notes Conversion

P Jayakrishna Sahit
jayakrishna.sahit@iitgn.ac.in
Indian Institute of Technology Gandhinagar
Gandhinagar, Gujarat

Rendla Aditya
rendla.aditya@iitgn.ac.in
Indian Institute of Technology Gandhinagar
Gandhinagar, Gujarat

Sai Praneeth Maddi
sai.praneeth@iitgn.ac.in
Indian Institute of Technology Gandhinagar
Gandhinagar, Gujarat

ABSTRACT

With electronic music and instruments being a growing trend, it is often easy to be able to convert anything we play on these instruments into MIDI files and be able to retrieve score cards for them, however for acoustic instruments, we must rely purely on the audio we capture using a microphone. Given that most of the research has been done on a piano, in this project we choose to present our research on data from an acoustic guitar. We wish to approach this by doing a comparative analysis on piano data and guitar data, and using these results propose a method to be able to retrieve musical notes from music of an acoustic guitar. We then evaluate our method by comparing them with methods which have been proposed exclusively on piano data.

KEYWORDS

midi, convolutional neural networks, Constant-Q transform(CQT)

ACM Reference format:
P Jayakrishna Sahit, Rendla Aditya, and Sai Praneeth Maddi. 2016. Music to Notes Conversion. In *Proceedings of ACM Conference, Washington, DC, USA, July 2017 (Conference'17),* 3 pages.
DOI: 10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Music identification from acoustic instruments is an added advantage for people interested in acoustic instruments, because they now have an easier way of learning songs using such instruments. On a broader note, understanding such instruments can give us better insight on understanding sound from non-electrical devices, such as the human voice box, etc. For a person who is a music enthusiast and has learned basics of music, music in the form of notes is very important. Since he/she cannot store notes of all the music he/she wants to learn, there should be some way to automate this. So, the main problem is to detect a bit of music which note it belongs to and we can achieve this by machine learning methods. Training the model on different notes at different instances(samples) and then predicting the note when a music bit is given. It is a supervised ML problem, with ground truth labels as musical notes of training data. This project is done as part of course Machine learning.

Conference'17, Washington, DC, USA

DOI: 10.1145/nnnnnnn.nnnnnnn

2 RELATED WORK

With the goal of becoming the ultimate platform for notation based music creation Doremirfis unique technology enables users to easier than ever before capture and digitize their musical creativity, just like "`Google Translate`", but for music. The technology behind this product is based on 20 years of research in music cognition scorecloud.com. Previous work on this topic had been done by Karey Shi, Michael Bereket and titled their paper as "`An AI Approach to Automatic Natural Music Transcription` ".

3 DATA COLLECTION

Primary data collection involved collection of MIDI files for various guitar songs. We were careful to make sure that these songs only had guitar notes and nothing else. After collecting about 500 songs, we converted the following songs into their mp3 versions. One major challenge for the following was the resources for pure guitar notes MIDI files and also the time it took for us to convert the MIDI files into their respective mp3 versions.

4 DATA ANALYSIS

The main part of our data analysis involved understanding the MIDI files. A major challenge we faced in this section of the project was the lack of good documentation for python modules which handled MIDI files. Although some of them did have good documentation such as Pymidi, dependency errors between python 2 and python 3 made it uneasy for all of us to work on it, thus usually forcing us to use a single person's machine because only it has the right settings.

An analysis of the MIDI files, does throw a good and important observation, which is that the notes present in these songs were present in only a certain bandwidth of notes and that the rest of the notes never appeared at all in any song. Most songs usually lasted for 4 minutes which made our dataset uniform in that aspect.

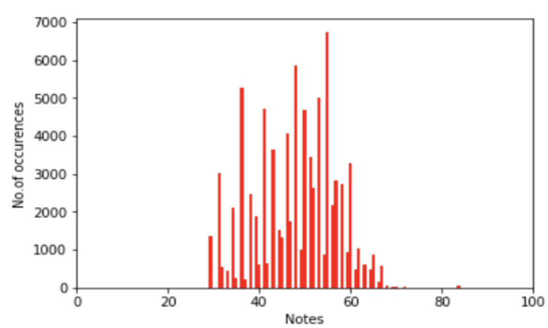

Figure 1: Notes distribution over songs

5 CHALLENGES

Apart from the data collections, we were faced with a number of challenges and key decisions which we had to take. The first one involves understanding methods for feature extraction from the mp3 files. Given an audio signal, using a waveform frequency transformation seemed to be good method, however choosing the transform would be important. Plots of these transforms could introduce new biases which could affect the out come of the machine learning algorithm.

5.1 Understanding other machine learning algorithms

The second one involved understanding which methods could be best used to help us solve this problem. Although previous work had been majorly done with the help of Convolution Neural Networks, it would be a major help if other methods could be combined with the following to create an ensemble approach to the following. Given the pool of machine learning algorithms we knew and rigorous testing, it was hard to identify such methods, which could help extract important features to learn from.

method	accuracy
svm	22.3%
logistic regression	17.8%
decision trees	34.2%
random forests	30.9%

Table 1: accuracy for different machine learning algorithms

5.2 State of the Art

Our given state of the art involved neural networks to classify notes. Thus it was important for us to first understand how our state of the art performed. The model consisted of 2 convolution layers followed by two dense layers. We were able to achieve the following results from it.

accuracy	98.72%
precision	0.6
recall	0.52
f1-score	0.55

Table 2: metrics for the state of the art

Thus given the following results there was a need for fine tuning the following model to our specifications

6 METHODOLOGY

The main method involves an important preprocessing procedure and our neural network architecture, which is a fine tuned version of the above described state of the art version. The basic outline of our model would be to, given a song extract the notes from this song. The number of notes is fixed to 88.

Data Preprocessing

Our preprocessing method included converting the mp3 files we had converted from our MIDI files as previously mentioned into a waveform plot. For this we used the Constant Q-transform. We then repeated the following process for all available mp3 files. These plots are our models input. An important task in this involves understanding the window of these images to be fed into the neural network. We decided to take a window size of 7 columns. This meant that a n x 7 would produce a time period from which we would predict the occurrence of the 88 notes. This time period is the unit time period considered for the song.

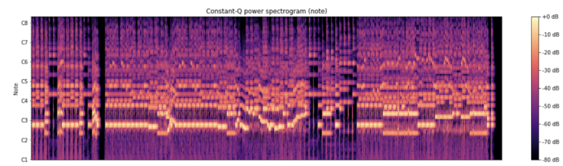

Figure 2: CQT plot of the song

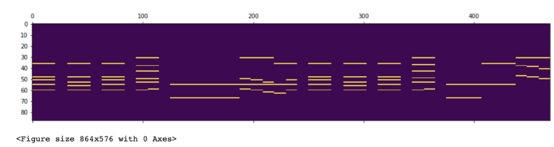

Figure 3: Notes plot of the song

While deriving truth values i.e, the notes of the songs from the MIDI files, it is made sure that the number of items taken across the x-axis on the plot is the same as the music notes plot, thus ensuring a match for both the number of images per song and number of notes being extracted. The notes for a 7 columns wide window was found out by considering a threshold value for that window. In our case the threshold was taken as 4.

Training our model

Our model is a slight tweak of the our state of the art method, with respect to the hyper parameters of the dense layers of the network.

We initially trained our model on 170 songs which comprised of about 4000 images along with truth values. We then increased our data-set to 500 songs in our data-set which gave us about 11,000 individual CQT images and their respective notes.

Predicting

We preprocess our given mp3 in the same way we did for our data-set and then pass it through our neural network. We then obtain the notes for each window and combine these to generate a music sheet for the following song.

7 RESULTS

After training our model we manage to achieve an accuracy of about 97% a precision of about 0.175, recall of about 0.225 and a

Figure 4: Music sheet generated from our model

total f1-score of about 0.2. The lower numbers in the precision and recall can be attributed to the high bias towards the number of zeroes in our training set.

accuracy	97%
precision	0.286
heightrecall	0.151
f1 score	0.2
Matthews correlation coefficient	0.2

Table 3: metrics for our trained model

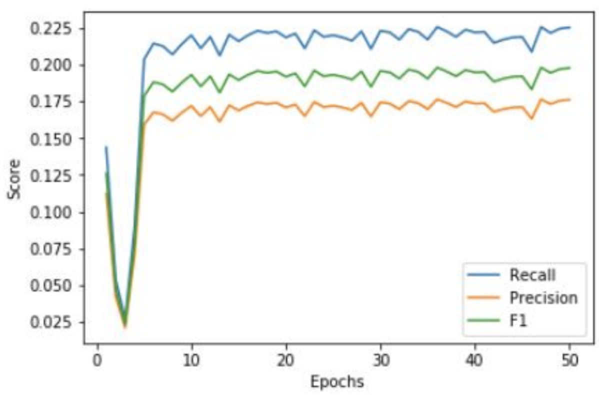

Figure 5: metrics vs epochs

8 CONCLUSIONS

Architectures used for keyboard music did provide good scores, however different ones were necessary. In a way it proves our hypothesis (not conclusively though) that guitar music is different from keyboard music. As future work to this project we wish to understand how the implications of these towards other instruments

ACKNOWLEDGMENTS

We would like to thank Prof.Nipun Batra and Supratim Shit for helping us in this project and for guiding us through it.

REFERENCES

[1] Michael Bereket, Karey Shi "An AI Approach to Automatic Natural Music Transcription" (2016).
[2] Martin Weiss Hansen, Mads Grsbll Christensen "Estimation of Fundamental Frequencies in Stereophonic Music Mixtures" (2016)
[3] McFee, Brian, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. filibrosa: Audio and music signal analysis in python.fi In Proceedings of the 14th python in science conference, pp. 18-25. 2015.
[4] Jayakrishna Sahit, Rendla Aditya, Sai Praneeth Maddi. Guitar MIDI files opened from "https://www.classicalguitarmidi.com/"
[5] Thomas A Caswell, Michael Droettboom, John Hunter, Eric Firing, Antony Lee, Jody Klymak, fi Elan Ernest. (2019, February 26). matplotlib/matplotlib v3.0.3 (Version v3.0.3). Zenodo. http://doi.org/10.5281/zenodo.2577644
[6] Vladyslav Sarnatskyi and Vadym Ovcharenko and Mariia Tkachenko and Sergii Stirenko and Yuri Gordienko and Anis Rojbi