Music Genre Classification Using Neural Networks

Shiraz Yaacob, Oran Goldrich

IDC, Israel, 2020

Abstract

With music being released on a daily basis over global platforms, the need for accurate songs criterion is rising rapidly. An automatic, simple way to achieve such classification is necessary functionality for many companies: Spotify, Apple Music, SoundCloud etc. We decided to use Deep Learning, and implemented CNN, RNN, and CRNN. We tested it over data-sets of classified songs, pre-processing them first into mel-spectrograms. We found out that song classification into genre is a rather complicated task when the instances are mel-spectrograms. While different architectures of CNN provided satisfying results, RNN functioned poorly. CRNNs achieved almost the same result as CNNs.

The Problem in More Details

As humans, detecting song genres is a rather simple task, but not for a machine. As expected, Music genre classification is not a new problem in Machine Learning. Many have tried to implement different algorithms to tackle it. Some works, such as [1], classified music genres according to tempo (BPM - Beats per Minute) using SVM. This work shows decent results with 67% accuracy over 8 genres. In recent years, using MFCC (Mel-frequency cepstral coefficients) is one of the leading approaches for music genre classification [2][3]. Top-notch models today achieved 91% accuracy over 10 genres [3]. We used FMA as a data-set- 8 GB of raw audio files [4], pre-processed by Priya Dwivedi [5]. The audio comes from more than 16,000 artists, and each song was cut to samples of 30 seconds. The FMA provides three subset of the full data-small, medium, and large, which include 8K, 25K and 106K of trimmed songs respectively. Since we can't process such large scale data-sets, we choose to use the small one.

Method

Our general approach to this classification problem has a few steps. In the first step, as we treat a spectrogram of a song as an image, we implemented different CNNs. Later on, as our data embodies a time dimension, we put a RNN into use. RNNs might be able to aid identifying the short and long term temporal features in the song. The last step was to combine the two into a CRNN attempting to take advantage of local feature extraction of CNNs with temporal summarization of a RNN. In all our models we used soft-max classifier. Also, we used L2 regularization to reduce over-fit.

Firstly, we implemented two 2D CNN models - A shallow and a deeper one. Our main line of thought focused on the fact that the complexity of our images is low. Unlike usual images, mel-spectrograms contain rather simple shapes. Following this line, we hypothesize that a shallow CNN model might have similar or better performance compared to a deeper model.

Secondly, we implemented two 1D CNN models - A shallow and a deeper one as well. We paid close attention to the fact that our input is a spectrogram rather than a regular image. Hence, we were not sure that 2D convolutional layers have any positive impact over 1D convolutional layers.

Then, we took the RNN hammer out of the toolbox. We decided to implement a naïve straight-forward RNN model

Lastly, it made perfect sense to combine the two using a CRNN model [6]. Such model introduces a convolutional network where the last convolutional layer is replaced by a RNN. We thought that this model might perform best.

We experimented further with our CRNN model by stacking LSTMs, different drop-out approaches and deepening the network.

Results

We ran each model for 70 epochs using a batch size of 32. For each model, we plotted train and validation accuracy and then plotted train and validation loss. Lastly, we calculated accuracy on the test set. In total, we built seven models to attempt verification of our hypotheses: CNN 2D Deep, CNN 2D Shallow, CNN 1D Deep, CNN 1D Shallow, RNN, CRNN, CRNN Stacked.

Below is a table of all models together, comparing accuracy performances and training time. Like before, the models have been numbered respectively: (1) CRNN, (2) CRNN Stacked, (3) RNN, (4) CNN 2D Deep, (5) CNN 2D Shallow, (6) CNN 1D Deep, (7) CNN 1D Shallow.

Model	(1)	(2)	(3)	(4)	(5)	(6)	(7)
Training Time (sec)	1738	3714	233	1182	321	673	526
Max Validation Accuracy (%)	51.875	51.875	14.75	57.625	54.0	53.75	54.375
Max Text Accuracy (%)	27.125	40.125	11.1	40.375	31.5	33.125	33.125

Below are two accuracy figures of our most successful models: Deep 2D CNN model and stacked CRNN model:

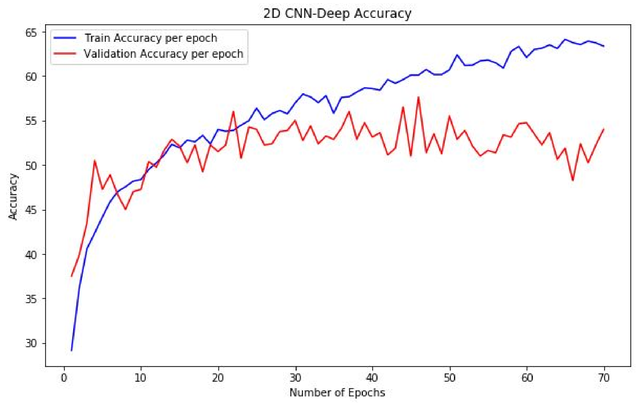

Fig. 1: CNN 2D Deep

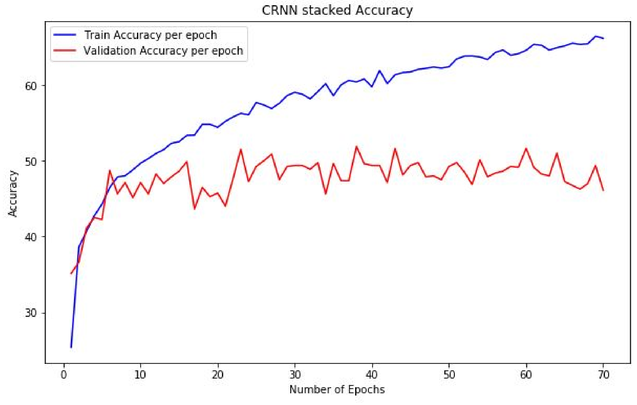

Fig. 2: CRNN Stacked

Conclusions

One of the simplest insights is the over-fit. After dozen epochs we reach over-fit. These results are easy to explain- as shown in our paper, the diversity of the 8 genres mel-spectrograms is not that different. Moreover, the complexity of the shapes in the mel-spectrograms is very low. Our models do not need many epochs to learn the pattern, so we reach over-fit fast.

For the same reason, our deep and shallow 2D CNNs show similar validation accuracy. Naïvely, we thought that since deeper networks can detect more complex patterns, it is more prone to over-fit. Surprisingly, the shallow model over-fits faster.

When we decided to run 1D CNNs as well, our main line of thought was that convolving over a single axis might produce better results. In practice, the validation accuracy remains pretty much the same. We observed that our 1D models produce less over-fit than the 2D models. Also, they introduce faster run-times.

Our RNN model performed poorly. We predicted this bad performance. Although dealing with a time dimension, our data is not really that sequential.

We had great expectations from our CRNN and stacked CRNN models. Our stacked LSTM achieved 40.125% test accuracy, and apparently generalizes better than the first LSTM model that achieved 27.125%. We thought that combining CNNs with RNNs will out-perform all other models. However, we did not manage to overtake the CNN models. Also, it is important to note that today's top-notch models in music genre classification avoid use of RNNs, such as [7]. These results probably emphasize the idea that mel-spectograms are far from being sequential data.

References

[1] Gouyon F., Dixon S., Dance Music Classification: A Tempo-based approach.
[2] M. Haggblade et al., Music Genre Classification.
[3] Y. Panagakis, C. Kotropoulos, and G. R. Arce. Music genre classification via sparse representations of auditory temporal modulations. In 2009 17th European Signal Processing Conference, pages 1–5, Aug 2009.
[4] Michaël Defferrard, Kirell Benzi, Pierre Vandergheynst, Xavier Bresson. "FMA: A Dataset For Music Analysis". url: https://arxiv.org/abs/1612.01840
[5] Priya Dwivedi, url: https://github.com/priya-dwivedi/
[6] Choi K. et al., Convolutional Recurrent Neural Networks for Music Classification
[7] Y. Panagakis, C. Kotropoulos, and G. R. Arce. Music genre classification via sparse representations of auditory temporal modulations. In 2009 17th European Signal Processing Conference, pages 1–5, Aug 2009.