International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

TAMIL MUSIC GENRE RECOGNITION WITH DEEP NEURAL NETWORKS

Hari Priya H[1], Harini C[1], Priyadharsini R[1], Mr. S. A. Gunasekaran.[2], Dr. J. B. Jona[2]

[1]*Student, M.Sc Decision and Computing Sciences, Coimbatore Institute of Technology, Coimbatore, India*

[2]*Associate Professor , Dept. Of Computer Applications, Coimbatore Institute of Technology, Coimbatore, India*

ABSTRACT

Music Genre Classification is one of the most prolific areas in the classification of machine learning techniques. The most popular classification methods for this is the use of deep learning techniques, most notably the Neural Networks (or NN) to process large music datasets to identify the corresponding genre. A well-known architecture has been used in the field and transfer learning techniques to adapt it to this task for the classification of The popular music genres in Tamil like Rock, Romantic, Melody, Classical, Devotional and Beats. Different strategies are used for fine-tuning, initializations and optimizers will be discussed to see how to obtain the model that fits better in the music genre classification. Finally, the performance is evaluate with a processed dataset for different Tamil genres.

Keywords: *CNN, Tamil music, Transfer Learning, Genre classification*

1. INTRODUCTION

Everyone has their own musical preferences, but this is the most fascinating aspect about music. Despite its versatility, music brings people together, heals, and calms, and favourite genres reveal a lot about the personalities. This integrates genre classifier with the help of deep learning. A m ic genre is an important feature that can direct users to their desired category. Many music people make playlists based on specific genres, which could lead to potential applications such as playlist recommendation and management. Despite previous studies on music genre classification using machine learning approaches, in which various algorithms were implemented and produced promising results, there is still room for improving genre classifier performance. In this paper, a music genre classification system is built using various machine learning techniques. The goal of this project is for this genre classifier to correctly classify a new music track which The datasets are collected on the basis of its associated features.The dataset was created in six distinct genres study's baseline. Error analysis is performed and future work is proposed based on the classification accuracy of each model. This is used for developing the music genre classification system and implementing additional system calibration, such as error analysis. It contains 600 tracks of music organised into 6 genres in a hierarchical taxonomy. Each track contains 30-sec audio tracks and is converted into wav format. These characteristics are obtained through data pre-processing of music tracks with the Python package. In this report, the efforts to develop classification methods are discussed that allows to identify a specific genre based on audio features. The dataset was created by cropping the songs. Then, to train the dataset, Support Vector Machine and softmax regression were tried with optimised parameters. CNN was used as the

Table 1: Categories in Dataset

	Genre	Count
1	Rock	100
2	Melody	100
3	Devotional	100
4	Romantic	100
5	Classical	100

6	Beats	100
	Total	**600**

2. LITERATURE REVIEW

Music Genre Classification is an area which has attracted the interest of many researchers. This section will provide details about some of the research work already done in this field. Vishnupriya S and K Meenakshi have proposed a Neural Network Model to perform the classification. Tzanetakis and Cook pioneered their work on music genre classification using machine learning algorithm. They created the GTZAN dataset which is till date considered as a standard for genre classification. Changsheng Xu et al have shown how to use support vector machines (SVM) for this task. Matthew Creme, Charles Burlin, Raphael Lenain from Stanford University have used 4 different methods to perform the classification. They have used Support Vector Machines, Neural Networks, Decision Trees and K-Nearest Neighbours methods to perform classification. Tao [5] shows the use of restricted Boltzmann machines and arrives to better results than a generic multilayer neural network by generating more data out of the initial dataset, GTZAN. After carrying out the above mentioned literature survey, Convolutional Neural Network is used to perform classification and the details of the same are explained in the following sections.

3. HANDMADE DATASET:

In this work, an Audio Set which is large-scale is used. Since the GTZAN genre classification dataset consists of the music of English Songs, a handmade dataset is created to classify the genres for Tamil. The videos were downloaded from YouTube into mp3 format. The mp3 files are converted into the desired wav format using an audio converter The dataset includes 6 different genres (Rock, Melody, Devotional, Romantic, Classical and Beats) with 100 songs per genre (each 30 second samples). Since they were all .wav files. The audiofiles cover classes of sounds including musical instruments, vocals etc.

4. METHODOLOGY AND WORKFLOW

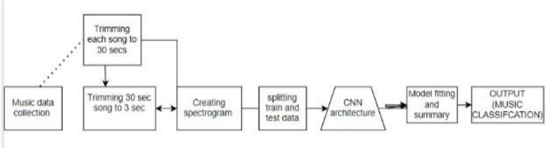

Trimming each song to 30 secs
Music data collection
Trimming 30 sec song to 3 sec
Creating spectrogram
splitting train and test data
CNN architecture
Model fitting and summary
OUTPUT (MUSIC CLASSIFCATION)

Fig 1: Workflow

This section provides the details of the data pre-processing steps followed by the description of the two proposed approaches to this classification problem.

4.1 Data Pre-processing:

The song that were collected consists of each genre comprising 100 audio files (.wav) of 30 seconds each implying 600 training examples and it is split into 20% for validation, which consists of 480 examples. Since, 30 seconds is little too much information for the model to take at once, a single audio file into 10 audio files each of 3 seconds. Now the training examples consist of 1000 training examples of each genre and total training examples are 6000.

4.2 Neural Networks:

The input layer of the Neural Network model reads in k features chosen using the forward model selection method. To achieve the best performance, the number of hidden layers were reduced to two, both of which use ReLu activation functions with 320 and 32 hidden units, respectively. The softmax function provides a probability distribution for multiple genre labels as the activation function for the output layer. A mini-batch size of 128 was used to train the model iteratively.

4.2.1 Spectrogram Generation

The CNN model needs an image as an input, for which the mel spectrograms of audio files are used and saved as an image file (.jpg or .png). This method has been widely used in several research papers on music genre classification. CNN works by taking several spectrograms extracted from audio files as inputs and modifying their patterns into a 2D convolutional layer with appropriate filter and kernel sizes. The spectrograms in CNN are used because of the model's ability to recognize image details accurately.

Implementing the Convolutional Neural Network (CNN) model using a pre-processed handmade audio tracks dataset. For each song, the dataset included an extracted Mel-frequency Cepstrum Coefficient (MFCC) spectrogram. In addition, the audio excerpts of 3 seconds and 30 seconds were accompanied by feature descriptions compiled in an additional document. CSV document. Feature shows how to extract vectors in Python using the librosa package. librosa is a Python package for tunes and audio analysis that provides the foundation for creating music information retrieval systems.

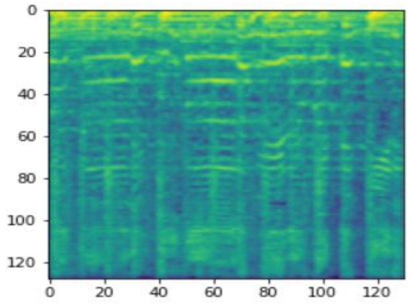


Fig 2 : Spectrogram sample

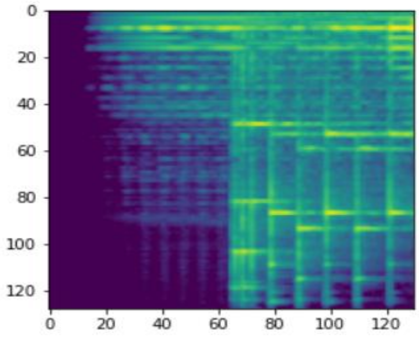


Fig 3 : Spectrogram sample

5. CNN FOR MUSIC CLASSIFICATION

The concept of constructing a 2D Convolutional Neural Network (CNN). Using the Librosa library, the audio files extracted into various types of spectrograms using the Audio tracks dataset. These spectrograms were used as binary inputs for a 2D CNN model built with the Keras library. TensorFlow was also used to create the layers. A 2D convolutional layer with an input shape of 128x128x1 was presented. It contained a 2D NumPy array containing the inputs, which would be passed to the max-pooling layer, which would then operate a matrix half the size of the input layer. There were 5 convolutional layers, each with a 2x2 kernel, a stride of 2, and a max-pooling layer.

6. PROPOSAL

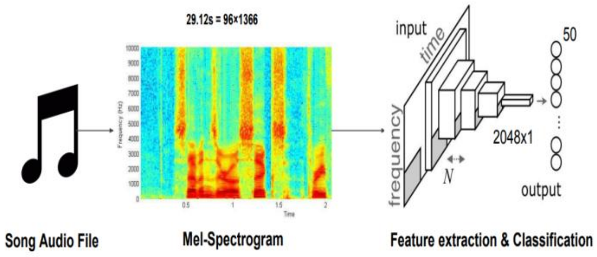


Fig 4 : Process Flow

6.1 Network architecture

In this work, the starting point will be the architectures of a CNN. The convolutional neural network consists of 5 convolutional layers of 33 kernels and max-pooling layers ((2 × 4)-(2 × 4)-(2 × 4)-(3 × 5)-(4 × 4)) as illustrated in Figure 1 a. The network reduces the size of feature maps to 1 × 1 at the final layer, where each feature covers the whole input. This model allows time and frequency in variances in different scales by gradual 2D sub-samplings. Being fully convolutional reduces considerably the number of parameters.

In its CNN substructure, the sizes of convolutional layers and max-pooling layers are 33 and (2 × 2)-(3 × 3)-(4 × 4)-(4 × 4). This sub-sampling results in a feature map size of N × 1 × 15 (number of feature maps x frequency x time). They are then fed into a 2-layer RNN, of which the last hidden state is connected to the output of the network.

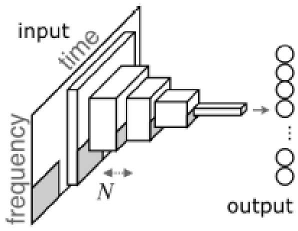


Fig 6 : CNN Architecture

6.2 Transfer Learning

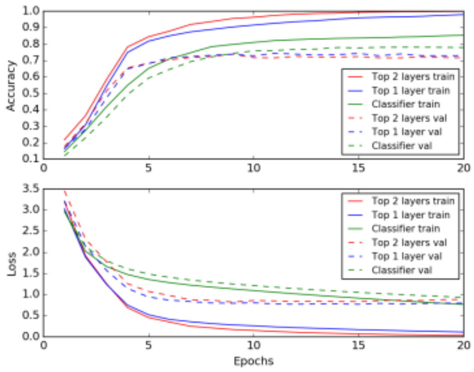
Transfer learning has proven to be very effective in the image processing scene, it studies and provides techniques on how to adapt a model trained in a large-scale database to perform well in other tasks different from the one that was trained so far. This paper aims at learning from a source data distribution (multiclass tags) a well performing model on a different target data distribution (single class genres). Inside the transfer learning paradigm this is known as domain adaptation. The two most common practices that is applied are: Using the network as a feature extractor. That is removing the last fully-connected layer and treating the network as a feature extractor. Once the features are extracted at the top it can include a classifier like SVM or a Softmax classifier for the new dataset. Fine-tuning the network. This strategy is based on not

only replace the classifier layer of the network, but also retrain part or the whole network. Through backpropagation the weights are modified and pre-trained the model to adapt the model to the new data distribution. Sometimes it's preferable to keep the first layers of the network fixed (or freezed) to avoid overfitting, and only fine-tune the deeper part. This is motivated because the lower layers of the networks capture generic features that are similar to many tasks while the higher layers contain features that are task and dataset oriented as demonstrated .

6.3 Multiframe

The propose to use a multiframe strategy that helps to extract more than one frame (or mel-spectrogram image) per song. Each song will discard the first and last N seconds and then, divide the rest into frames of equal time-length t. The final parameters are stated in the experiments. This approach has two advantages: At training time: this will generate more data to train the network than in the approach of, as they are only extracting the central part of the song. There are very limited by the number of songs, this is a very useful tool to provide data augmentation. At test time: this can average or perform a KNN with the scores of every frame to infer the genre tag for the complete song with more confidence.

7. EXPERIMENTS



In one frame 29.12s per song is extracted.

Fig 6 : Fine-tuning results of architectures

Specifically, this frame contains the central part of the song as it should be the most representative. Then, a log-amplitude mel-spectrogram is extracted using 96 mel-bins and a hop-size of 256 samples, resulting in an input shape of 96 × 1366. In reference to the multiframe approach, the song is divided in a set of frames of also 29.12s. Therefore, at each frame a mel-spectrogram can be extracted using the same parameters, and with the same resolution. In order to select the 29.12s that compound each frame, two different steps are carried out. Firstly, a short period of the song is removed both in the beginning and in the end. These parts of the songs are hardly ever representative. Therefore, their inclusion to the classification procedure would lead to weaker results. The following step consists in dividing the remaining part of the song in frames of 29.12s. They are not overlapped and the last frame is also removed if its duration is lower than 29.12s. Finally, once obtained all the frames, the evaluation criterion can be set at frame level or at song level. The accuracy metric is used to measure the performance of our system. If evaluated at song level, it is proposed to use an averaging of the predicted tags for each frame of the song. In order to do that, the mean among the tag scores of all the frames of the song is computed and the tag with the highest score is selected. Therefore, if a song contains a small period that can be classified as a different genre, it will not affect the final song classification. Another approach that would lead to a unique tag per song would be the nearest neighbours algorithm between the tags of all the frames of the song. Therefore, the most repeated tag among all the frames of the song would be selected as the tag of the song.

8. CONCLUSION

To explore the application of CNN and CRNN for the task of music genre classification focusing in the case of a low computational and data budget. The results have shown that this kind of networks need large quantities of data to be trained from scratch. In the scenario of having a small dataset and a task to perform, transfer learning can be used to fine-tune models that have been trained on large datasets and for other different purposes. The MultiFrame approaches are shown with an average stage improves the single-frame song model. In the experiments, a homemade dataset compounded by songs longer than our frame duration has been used. These songs belong to 6 different genres of Tamil and the experiments have revealed that the average stage achieves better results with higher total accuracy. Therefore, using the average stage the non-representative frames dependence can be removed. As a future work, some other techniques to obtain a single genre tag per song from multiple frames can be analysed.

REFERENCES

[1] music audio feature learning," in 14th International Society for Music Information Retrieval Conference (ISMIR-2013). Pontif´ıcia Universidade Catolica do Paran ´ a, 2013, pp. 116–121. ´

[2] Aaron Van Den Oord, Sander Dieleman, and Ben- ¨ jamin Schrauwen, "Transfer learning by supervised pre-training for audio-based music classification," in Conference of the International Society for Music Information Retrieval (ISMIR 2014), 2014.

[3] Sander Dieleman and Benjamin Schrauwen, "End-toend learning for music audio," in 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2014, pp. 6964–6968.

[4] Keunwoo Choi, George Fazekas, and Mark Sandler, "Automatic tagging using deep convolutional neural networks," arXiv preprint arXiv:1606.00298, 2016.

[5] Keunwoo Choi, George Fazekas, Mark Sandler, and Jeonghee Kim, "Auralisation of deep convolutional neural networks: Listening to learned features," in Proceedings of the 16th International Society for Music Information Retrieval Conference, ISMIR, 2015, pp. 26–30.

[6] Paulo Chiliguano and Gyorgy Fazekas, "Hybrid music recommender using content-based and social information," in 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016, pp. 2618–2622.

[7] Aaron Van den Oord, Sander Dieleman, and Benjamin Schrauwen, "Deep content-based music recommendation," in Advances in Neural Information Processing Systems, 2013, pp. 2643–2651.

[8] Keunwoo Choi, George Fazekas, and Mark Sandler, "Explaining deep convolutional neural networks on music classification," arXiv preprint arXiv:1607.02444, 2016.

[9] Duyu Tang, Bing Qin, and Ting Liu, "Document modeling with gated recurrent neural network for sentiment classification," in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, pp. 1422–1432.

[10] Zhen Zuo, Bing Shuai, Gang Wang, Xiao Liu, Xingxing Wang, Bing Wang, and Yushi Chen, "Convolutional recurrent neural networks: Learning spatial dependencies for image representation," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2015, pp. 18–26.

[11] Siddharth Sigtia, Emmanouil Benetos, and Simon Dixon, "An end-to-end neural network for polyphonic piano music transcription," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 24, no. 5, pp. 927–939, 2016.

[12] Keunwoo Choi, George Fazekas, Mark Sandler, and Kyunghyun Cho, "Convolutional recurrent neural networks for music classification," arXiv preprint arXiv:1609.04243, 2016.

[13] Kyunghyun Cho, Bart Van Merrienboer, Dzmitry Bah- ¨ danau, and Yoshua Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," arXiv preprint arXiv:1409.1259, 2014.

[14] Sinno Jialin Pan and Qiang Yang, "A survey on transfer learning," IEEE Transactions on knowledge and data engineering, vol. 22, no. 10, pp. 1345–1359, 2010.

[15] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in neural information processing systems, 2012, pp. 1097–1105.

[16] Matthew D Zeiler and Rob Fergus, "Visualizing and understanding convolutional networks," in European Conference on Computer Vision. Springer, 2014, pp. 818– 833.

[17] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 806–813.

[18] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson, "How transferable are features in deep neural networks?," in Advances in neural information processing systems, 2014, pp. 3320–3328.

[19] George Tzanetakis and Perry Cook, Manipulation, analysis and retrieval systems for audio signals, Princeton University Princeton, NJ, USA, 2002.

[20] Spotify, "Spotify genres, the full listing," http://news.spotify.com/us/2009/03/ 24/spotify-genres-the-full-listing, 2009.

[21] Diederik Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.

[22] Leon Bottou, "Large-scale machine learning with ´ stochastic gradient descent," in Proceedings of COMPSTAT'2010, pp. 177–186. Springer, 2010