Project Report

Identifying Name of the Singer from Audio Data

by

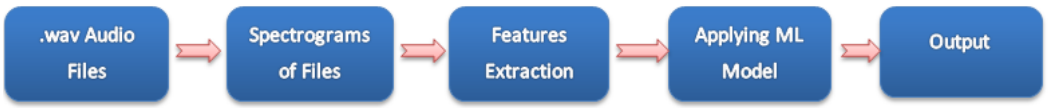
Group 11

Aarti Chandrakant Pol (M20MA002)
Anurag Saraswat (M20CS066)
Parsa Revanth (M20CS058)

Submitted to
Dr. Mayank Vatsa
Professor
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY, JODHPUR
February 14, 2021

1 INTRODUCTION

As the name suggests the aim of this project is to predict the singer based on his audio sample. We have built a deep learning model for this task. For performing audio classification[?][1], we have used the ImageNet-Pretrained standard deep CNN model as this acts as a strong baseline. We studied that the pre-trained weights are useful for learning spectrograms than using randomly initialized weights based on few papers. Then we performed a transfer learning-based approach. The basic block diagram for our model is shown below:



2 CONCEPTS RELATED TO AUDIO DATA

2.1 Spectrogram

For performing audio classification using CNN, we need to extract the features of the audio signal which is a non-periodic signal. These features can be extracted when the audio signal is converted into images. So, for this we plot spectrogram for the audio signal. Spectrogram is a visual representation of the spectrum of frequencies of audio signal as they vary with time. The axes of spectrogram represents time and frequency. The frequency represents pitch or tone of audio signal. The intensity or color is represented by third axis which is amplitude. Dark colors represent low amplitude and bright colors represent loud or strong amplitude. Below figure represents audio signal and the respective spectrogram.

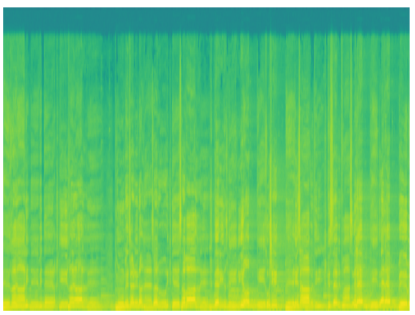


Fig. 1. Spectrograms

2.2 MFCC

MFCC is mel frequency cepstral coefficients.To further improve on the cepstral representation, we can include more information about auditory perception into the model. This can be done by introducing information about human perception, we focus the model on that part of the information which human listeners would find important. The log-spectrum already takes into

account perceptual sensitivity on the magnitude axis, by expressing magnitudes on the logarithmic-axis. The other dimension is then the frequency axis. MFCC preserves information and remove ”unrelated” information such as the pitch.

2.3 Mel-Spectrogram

Our eyes are better at detecting differences in lower frequencies than higher frequencies. So the frequencies are converted to mel scale such that the scale of pitches (high or low) can be judged easily because in mel scale the sound or pitches are equidistance from one another. So the Mel scale is a scale that relates the perceived frequency of a tone to the actual measured frequency. It scales the frequency in order to match more closely what the human ear can hear (humans are better at identifying small changes in speech at lower frequencies). Mel spectrogram is spectrogram with mel scale. In order to visualize the spectrogram, the y-axis is converted to log scale and amplitude axis is converted to decibels[4].

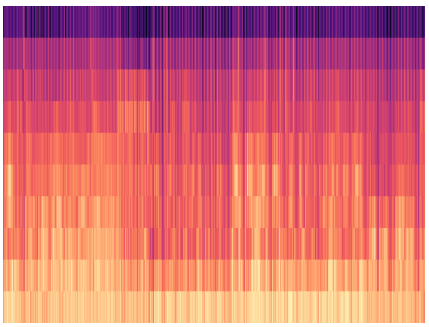


Fig. 2. Mel-Spectrogram

3 LIBRARIES USED IN THE PROJECT

3.1 Pydub

We used this library to convert the audio from mp3 format to wav format.

3.2 Keras Tensorflow

We used this for processing the spectrograms. Also used for importing the pretrained CNN models namely VGG16, ResNet50, and InceptionV3.[2]

3.3 Scikit-learn

We used this library to calculate the confusion matrix, precision, recall for the test data. Also used to import XGBoost classifiers.

3.4 Matplotlib

We used this library to visualise the data like spectrogram, mel-sprectrogram.

4 DATASET COLLECTION AND PREPROCESSING

We downloaded data from various online sources. We created a dataset of 10 artists namely Arijith, Shreya Ghoshal, Kishore Kumar, Kumar Sanu, Lata Mangeshkar, Mukesh, Sonu Nigam, Hariharan, Mohammad Rafi and Mohit Chauhan with each artist having 50 songs except Hariharan has 40 songs. The downloaded files are in mp3 format. For processing the audio data the data should be in wave format(.wav). We firstly convert the mp3 audio data to wav format data. After that for each audio sample we create the spectrograms and mel-spectrograms for a time intervel of 30 seconds starting from 30 seconds and we took 4 to 6 such spectrograms and mel-spectrograms based on the length of the audio file.

5 CNN MODELS USED FOR FEATURE EXTRACTION

Keras provides set of deep learning models which are pre-trained on weights of ImageNet dataset. So have used VGG-16, ResNet50 and InceptionV3 models for feature extraction.[3]

5.1 VGG16

VGG16 model has 16layers that have weights. The convolution layers used are having 3x3 filter with a stride 1. This model uses same padding and maxpool layer of 2x2 filter having stride of size 2. In the end it has 2 FC(fully connected layers) followed by a softmax for output. This network is a pretty large and it has about 138 million (approx) parameters.

5.2 InceptionV3

Inception model consist of 1x1 Convolutional layer, 3x3 Convolutional layer, 5x5 Convolutional layers. Their output filter banks are concatenated into a single output vector forming the input of the next stage. A typical Inception network can have several Inception layers and with occasional max-pooling layers with stride 2 in order to halve the resolution of the grid.

5.3 ResNet50

ResNet model consists of several residual blocks which are stacked on top of each other. These residual blocks has two 3x3 convolutional layers with the same number of output channels. Each convolutional layer is followed by a batch normalization layer and a ReLU activation function. A skip connection is added which skips these two convolution operations and adds the input directly before the final ReLU activation function.

6 XGBOOST CLASSIFIER

After feature extraction, we gave them as input to XGBoost Classifier. Boosting is a type of ensemble learning, it is used to enhance the performance of ML models. It is a process that uses a set of Machine learning algorithms to combine weak learners to form strong learners in order to increase the accuracy of the model. So these models are built sequentially by minimizing the errors from previous models while increasing or boosting performance of model. Gradient boosting uses a gradient descent algorithm to minimize the errors in sequential models. In our project we have used XGBoost which is a decision-tree-based ensemble ML algorithm which uses gradient boosting. So after doing feature extraction we pass them to the XGBoost classifier.

7 PROJECT SUMMARY

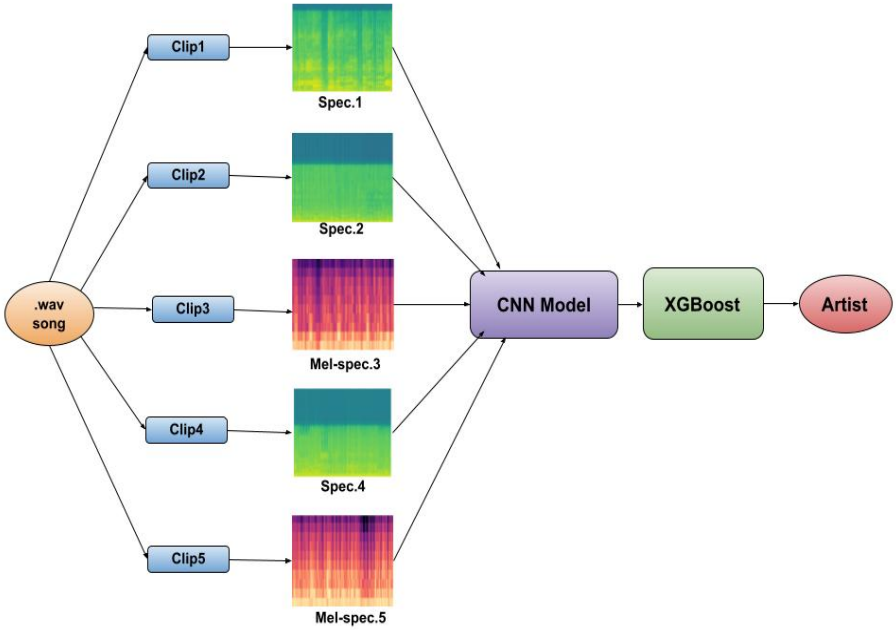


Fig. 3. Summary of Project

(1) A audio sample in .wav format is taken as input.
(2) We generate clips of the audio sample such that each clip is of 30 seconds.
(3) Depending on the length of the audio data, clips will be within range of 4 to 6.
(4) We generate spectrogram and mel spectrogram for each clip.
(5) Now, we combined spectrogram and mel-spectrogram for different clips.
(6) We pass the combined data into CNN model. Here we used three different CNN models VGG16, ResNet50 and InceptionV3 which are pretrained on imagenet dataset.
(7) We extract the features from the CNN model and pass it to XGBoost classifier.
(8) We get the probability for each sample and maximum probability class will be predicted as the output class.

8 PERFORMANCE

(1) Confusion matrix is used to have a complete picture while assessing the performance of model accuracy, precision and recall are used commonly to metrics for model evaluation.
(2) Precision tells what proportion of identifications for a label was actually correct.
(3) Recall tells what proportion of an actual label was identified correctly.

8.1 Confusion matrix

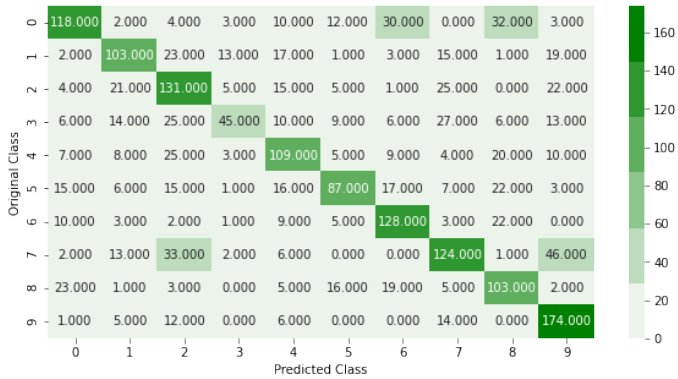


Fig. 4. Confusion Matrix for VGG16

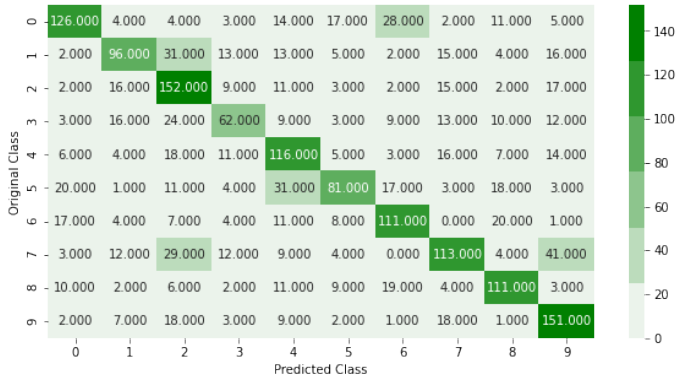


Fig. 5. Confusion Matrix for InceptionV3

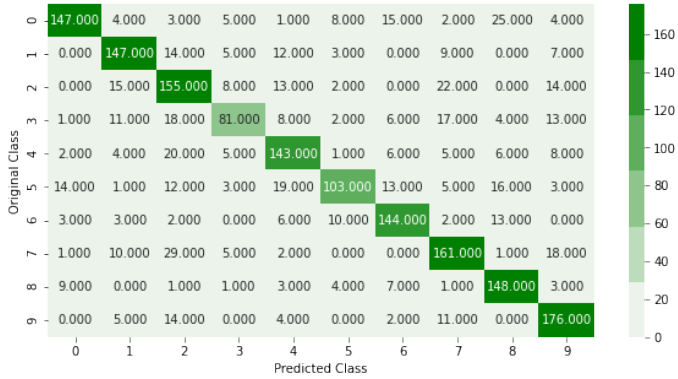


Fig. 6. Confusion Matrix for ResNet50

- From the above confusion matrix we can clearly see that when we use ResNet50 pretrained model we get the maximum numbers on the diagonal compared to InceptionV3 and VGG16.
- The model is getting confused with the singers Kishore kumar, Mukesh when we use VGG16 model.
- But this confusion fades away when we use model trained with InceptionV3 and it gets better when we use model trained with ResNet50.

8.2 Recall Matrix

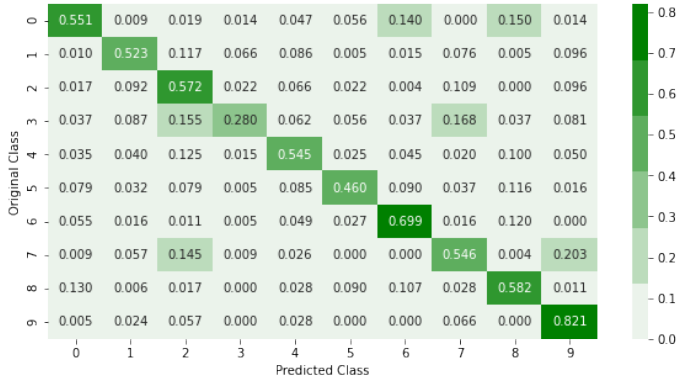


Fig. 7. Recall Matrix for VGG16

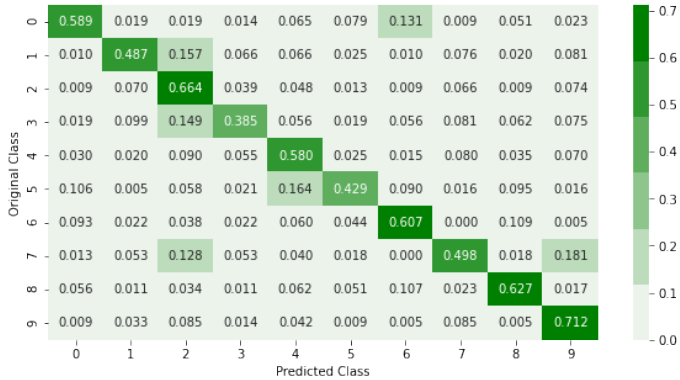


Fig. 8. Recall Matrix for InceptionV3

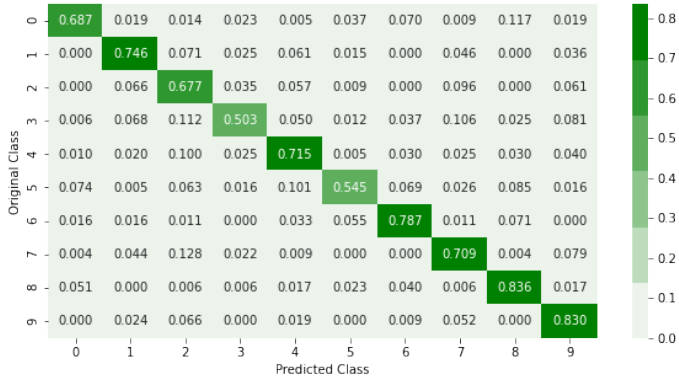


Fig. 9. Recall Matrix for ResNet50

8.3 Precision Matrix

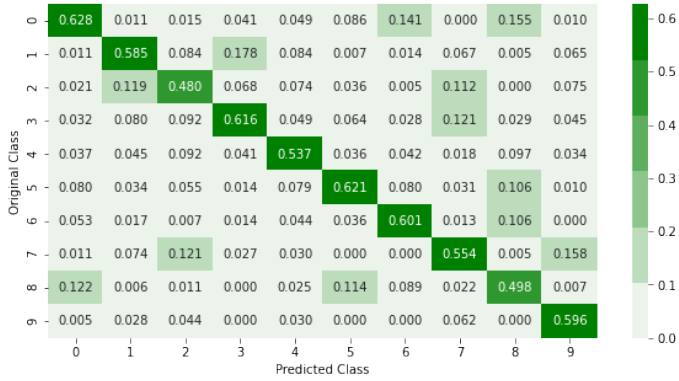


Fig. 10. Precision Matrix for VGG16

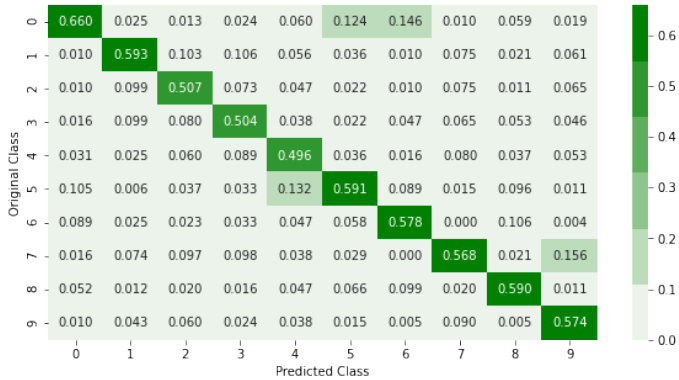


Fig. 11. Precision Matrix for InceptionV3

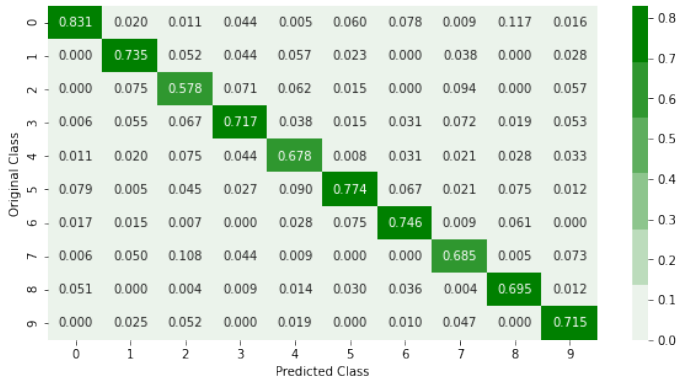


Fig. 12. Precision Matrix for ResNet50

8.4 Accuracy

Top-N accuracy means that the correct class gets to be in the Top-N probabilities for it to count as "correct" so calculated the accuracies for Top-1, Top-2 and Top-3 predictions.

S.NO	Top-1	Top-2	Top-3
VGG16	56.41%	76.27%	86.88%
InceptionV3	56.26%	76.02%	85.41%
ResNet50	70.64%	84.82%	93.07%

9 RESULTS

Example 1:

(1) Song Name : Woh Din and artist : Arijit Singh of an audio file.
(2) The top-3 predicted output for this file ['Arijit', 'MohitChauhan', 'ShreyaGhoshal']
(3) For this example model predicted correctly.

Example 2:

(1) Song Name : Abhi Mujh Mein Kahin and artist : Sonu Nigam of an audio file.
(2) The top-3 predicted output for this file is ['MohitChauhan', 'SonuNigam', 'Hariharan']
(3) For this example first prediction is wrong but the second prediction is correct.

10 CONCLUSION

We created a simple model which will extract features from CNN architecture and perform classification using XG-Boost. We achieved accuracy of 70% using pre-trained Resnet and over 93% including top3 prediction. There are also some mis-predictions due to some similarity between the frequencies of voices for some artist in few songs. Although this model performs rather well, but if you want to increase the accuracy then you can include more examples for each artist to train your model and do some feature selection to avoid overfitting before feeding them into the model. Above model can be extended to create backend of UI based application and can be extended to drag and drop based output application.

REFERENCES

[1] Jurgen Arias. [n.d.]. Voice Classification with Neural Networks. ([n. d.]). https://towardsdatascience.com/voice-classification-with-neural-networks-ff90f94358ec

[2] Asad Mahmood. [n.d.]. Audio Classification with Pre-trained VGG-19 (Keras). ([n. d.]). https://towardsdatascience.com/audio-classification-with-pre-trained-vgg-19-keras-bca55c2a0efe

[3] Kamalesh Palanisamy, Dipika Singhania, and Angela Yao. 2020. Rethinking CNN Models for Audio Classification. *arXiv e-prints*, Article arXiv:2007.11154 (July 2020), arXiv:2007.11154 pages. arXiv:2007.11154 [cs.CV]

[4] Arun Solanki and Sachin Pandey. 2019. Music instrument recognition using deep convolutional neural networks. *International Journal of Information Technology* (2019), 1–10.