Electronic Music Sub-genre Classifier

Authors
Shelby Neal (shelbyneal)
Daniel Auerbach (dauerbach99)

Abstract

This paper presents the progress made on creating an Electronic Dance Music (EDM) sub-genre classifier. The overall goal of the project is to build a classifier that will let a user know what sub-genre a given EDM song belongs to. This sub-genre classifier expands upon the current body of research on general genre classification. Sub-genre classification is important because as music genres grow and new sub-genres arise, listeners will need better models to help them find music they enjoy in genre niches. This project set out to determine the best way to classify EDM sub-genres. The dataset used consisted of 300 ten second clips of EDM songs with 50 songs from each of the six sub-genres being classified (Ambient, Drum and Bass, Dubstep, Hardstyle, House, and Trap). The data were evenly split among genres into 250 samples for training and 50 samples for testing. Pre-processing and normalization of the data was completed and a data frame was created to include features of the samples as they relate to classification. The project compared four different classification models including k-nearest neighbors, Gaussian Naive Bayes, decision trees, and support vector machines (SVM). The SVM model also tested four different kernels (linear, radial basis function, polynomial, and sigmoid). The scikit-learn python package was used to create and test each model. The k-nearest neighbors model with $k = 5$ and the decision tree model both performed the best overall with a classification accuracy of 68%. The worst performing model was the SVM with a polynomial kernel with an accuracy of 44%. A potential current limitation could be the number of audio samples, audio data compression from downloading, and the length of each sample.

1 Introduction

Music genre classification is an important topic in machine learning. Popular music streaming applications like SoundCloud and Spotify use genre classification to sort music to better meet the needs of their users. Current approaches to music genre classification are coarse grained by sorting between mainstream genres (Huang et al., 2018). Coarse grained classification is helpful for general sorting of music files, but with the music industry constantly dividing genres into sub-genres, the coarse grained approach may not be sufficient for specific sub-genre classifications. Finer grained music genre classifiers will help with music streaming services and statistical analyses of musical data.

To approach the problem of trying to classify sub-genres, a few sub-genres of electronic dance music (EDM) were analyzed. EDM generally features a rolling bass line with strong kick drums and lots of synthesized sounds. Additionally, many EDM songs are in a similar musical key and can thus have similar tonality. EDM contains over 20 genres that are further broken down into sub-genres. Six EDM sub-genres were used to gauge how well various machine learning models could discern between songs that have similar sounds. For this report, the data used were gathered from Soundcloud.com and pre-processed. The data consisted of samples of EDM songs from six different sub-genres. The data was then split into testing and training partitions and a few tests were run using seven machine learning models: k-nearest neighbor, linear kernel SVM, radial basis function kernel SVM, polynomial kernel SVM, sigmoid kernal SVM, Gaussian Naive Bayes, and a decision tree classifier. The models were trained on the same data to compare prediction accuracy for the sub-genres.

Each model tested was fine tuned to use the optimal hyperparameters before comparing with other models. K-nearest neighbors, for example, used five nearest neighbors since five was the k value which produced the highest accuracy. Each model also underwent five fold cross validation with the data. Humans can classify genres successfully for a four genre classifier at an accuracy rate of around 63% (Adragna et al., 2018), and humans can classify 10 genres with an accuracy of 70% (Dong, 2018). Given that this project is looking at six sub-genres to classify, it can be concluded that the models that performed well are likely on par or better than humans classifying EDM sub-genres.

2 Data

The data consists of 300 10-second samples of EDM songs, split into 250 samples for training and 50 samples for testing. To split the data into training and testing portions, the sklearn train_test_split() function was used. These 300 samples were evenly divided among the six sub-genres: 50 songs each from Ambient, Drum and Bass, Dubstep, Hardstyle, House, and Trap sub-genres. To select which 10 seconds to use from each song, the songs were first normalized within Audacity to compensate for variation in gain levels. Then, a 10 second portion from each song was chosen that best encapsulates its style (i.e., refraining from selecting an insignificant intro or outro).

The DataFrame was then created to include metrics significant for genre classification: the means and variances for chromatic short-time Fourier transformation, root-mean-square energy, Mel spectogram, spectral centroid/bandwidth/rolloff, zero crossing rate, harmonics, percussion, and tempo. To populate the DataFrame with data on these features for each sample, the Librosa package was used. Once the DataFrame was created, the feature data was normalized using sklearn's StandardScaler to fit_transform() the features to resemble a normally distributed dataset.

3 Machine Learning Models

To train the model, a number of machine learning algorithms were tested and confirmed using cross-validation. Initially, the k-nearest neighbors algorithm was used and the best performing 5-nearest-neighbors model categorized an audio sample into the correct sub-genre 68% of the time. Variances in the success rate were encountered based on the number of nearest neighbors and the distance metric used (Euclidian distance, Manhattan distance, or Minkowski distance), as well as which samples were selected for training.

Next, tests were performed with support vector machines (SVMs) using various kernel methods: linear, radial basic function, polynomial, and sigmoid kernels. After cross-validation, classification accuracy yielded for each of the kernel methods was, on average, 62%, 66%, 44%, and 58%, respectively. Also, a model was generated using the Naive Bayes' algorithm which yielded 58% accuracy. From these results, it was concluded that the SVM and Naive Bayes implementations were less efficient than the 5NN model.

Finally, a model was generated using a decision tree classifier with various max depths. After multiple runs and cross-validation, the decision tree generated with a max depth of 5 nodes performed with the highest accuracy of 68%; thus, tying the 5NN model in efficiency.

4 Results and Analysis

In discovering why audio samples were being misclassified, confusion matrices were generated for each of the models. After examining each sub-genre independently, the Ambient sub-genre was predicted perfectly in 4 of the models tested. This is due to the vast difference in energy levels for ambient sound when compared to other forms of EDM. Ambient sound generally consists of much softer sound sequences; thus, classification for this sub-genre was not an issue.

For the Drum and Bass sub-genre, there were minimal errors in classification which is explained by the fast-paced nature of Drum and Bass sound. Songs of this sub-genre tend to lean more towards 175 beats-per-minute which is much faster than the average tempo for the other sub-genres (70-150bpm). Additionally, very few errors were encountered when classifying songs from the Hardstyle sub-genre.

Hardstyle is known for pushing kick drum sound levels to extremes and overdriving them onto an offbeat bass sequence. Since the overall sound space for Hardstyle is also vastly different from its competing sub-genres, it was much easier to accurately classify this style.

The bulk of classification errors in the experiment came from the Dubstep, House, and Trap sub-genres, with highest error seen in Dubstep and Trap. This is explained my the many similarities shared among these sub-genres in modern EDM. Trap is a relatively new genre which originated in the early 2010s which blends elements of hip hop with buildups, drops, and breakdowns (styles predominantly used in Dubstep and House). Dubstep and Trap also tend to lie within the same tempo range (70-75bpm, 140-150bpm) versus the tempo range in House music (120-130bpm). It followed that the tempo variation in House music allowed for slightly better results than the other two poorer-performing sub-genres.

In conclusion, many different factors contributed to the overall efficiency of the various models tested. With a classification accuracy of 68% being on par with industry-standard for human-classified genre identification, the only question remaining is how to improve the model for future analyses. With samples of greater length and quality, more data could be added to the training model to improve results. Also, a secondary study could be performed on the overall correlation between these sub-genres. For example, since Trap music was born from Dubstep, there was higher collinearity between the two sub-genres. Due to this, testing this sub-genre in a binary model rather than a multiclass model would likely yield more accurate results. In future studies, longer song samples of higher qualities could also be used to improve accuracy. Regardless, this project was a good step in looking into the separation of musical genres with similar sounds using machine learning.

5 Citations, figures, tables, references

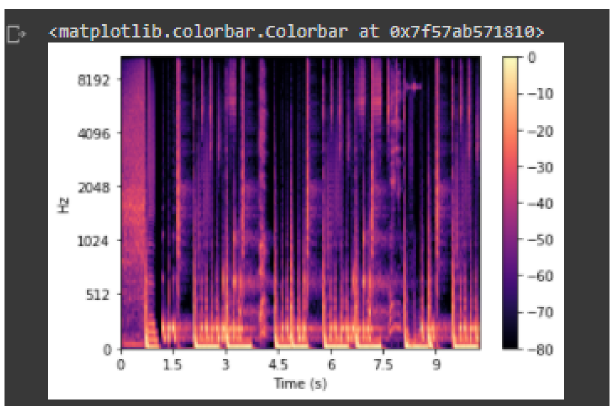

Figure 1: Feature Example: Mel-Scaled Spectrogram

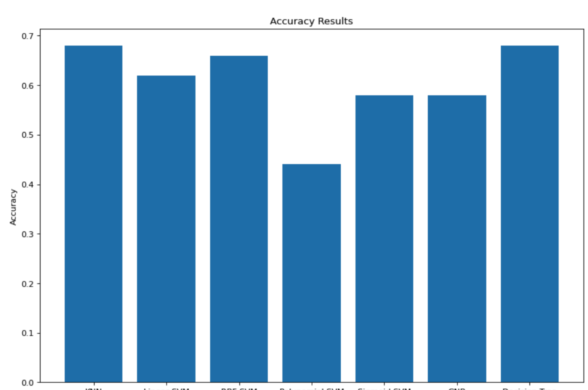

Figure 2: Comparison of Algorithm Performance

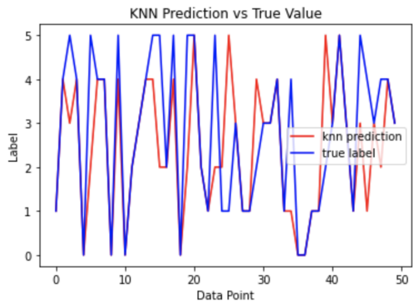

Figure 3: KNN Vs. Truth

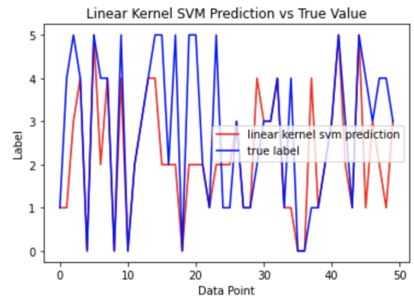

Figure 4: Linear SVM Vs. Truth

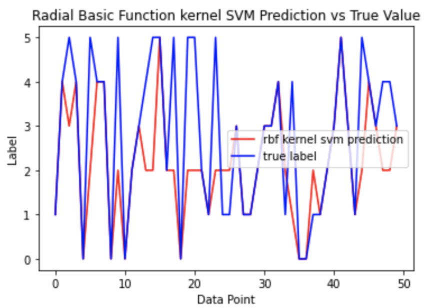

Figure 5: RBF SVM Vs. Truth

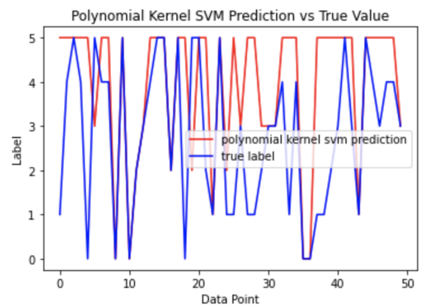

Figure 6: Polynomial SVM Vs. Truth

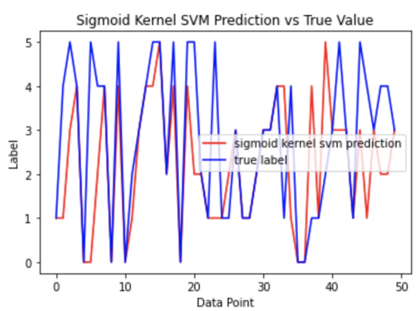

Figure 7: Sigmoid SVM Vs. Truth

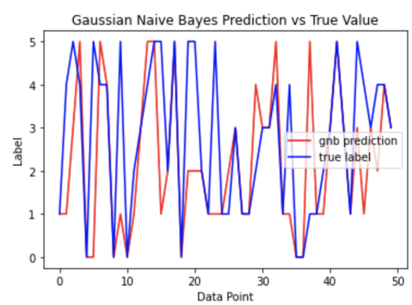

Figure 8: GNB Vs. Truth

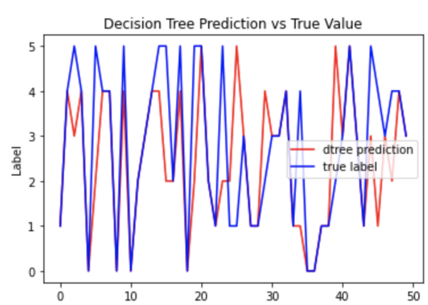

Figure 9: Decision Tree Vs. Truth

Table 1: Confusion Matrices for Best-Performing Models

Confusion Matrix (5NN)						
Ambient	6	0	0	0	0	0
Drum & Bass	0	8	1	0	0	1
Dubstep	0	0	3	0	1	1
Hardstyle	0	0	0	8	0	0
House	0	2	1	0	7	0
Trap	0	0	4	2	3	2

Confusion Matrix (D-Tree)						
Ambient	6	0	0	0	0	0
Drum & Bass	0	9	0	0	0	1
Dubstep	0	0	3	0	0	2
Hardstyle	0	0	1	7	0	0
House	0	1	1	0	6	2
Trap	0	1	3	4	0	3

6 Contributions

Shelby: Collected the sound files from SoundCloud and (using human-classified labels) organized the sound archive. Normalized and trimmed each sound file in Audacity to a 10-second standard and shared them with the group. Helped implement and test the various machine learning algorithms used during research. Worked evenly on the final report and presentation of experimental findings.

Daniel: Mounted audio files into Google colab notebook for pre-processing. Wrote code to pre-process data for training by collecting all hyperparameters for each audio file. Created and exported csv file of dataframe for ease of use in shared drive. Generated plots and graphs to help visualize result data. Worked equally on the final project report.

7 References

[1]Adragna, R., &; Sun, Y. H. (n.d.). Music Genre Classification. eecg.utoronto.ca. Retrieved October 2021, from https://www.eecg.utoronto.ca/ jayar/mie324/musicgenre.pdf.

[2]Chowdhry, A. (2021, May 7). Music genre classification using CNN. Medium. Retrieved October, 2021, from https://blog.clairvoyantsoft.com/music-genre-classification-using-cnn-ef9461553726.

[3]Dong, M. (2018, February 27). Convolutional neural network achieves human-level accuracy in music genre classification. arXiv.org. Retrieved December 8, 2021, from https://arxiv.org/abs/1802.09697.

[4]Huang, D. A., Serafini, A. A., &; Pugh, E. J. (2018). Music Genre Classifier. cs229.stanford.edu. Retrieved October 2021, from http://cs229.stanford.edu/proj2018/report/21.pdf.