Spotify Music Genre Classification

Audio and Music

1st Huseyin Emre Aksoy
Computer Engineering (of Aff.)
TOBB University of Economics and Technology
Ankara, Turkey
h.emreaksoy@gmail.com

2nd Hira Cansu Degirmenbasi
Computer Engineering
TOBB University of Economics and Technology
Ankara, Turkey
hiracansudegirmenbasi@gmail.com

***Abstract*—Many music listeners use the Spotify application and create thousands of playlists on a daily basis. Many users want to group songs by genre when creating these playlists, but the Spotify application does not provide information about the genres to which the songs belong. As a result, in this project, we aim to detect the genre and mode of Spotify songs and playlists. By doing so, we hope to provide users with a more intuitive and convenient way to organize their music by genre. This will allow them to more easily find and listen to the types of music that they enjoy, and to create more diverse and dynamic playlists.**

***Index Terms*—Music, genre, music classification, machine learning, SVM, Random Forest, Logistic Regression, Naive Bayes**

I. Introduction

The music genre is a key feature of any song that can help users to find and discover new music that they will enjoy. Many music streaming platforms, such as Spotify, allow users to create playlists that include specific genres of music, making it easier for users to find and listen to the music that they love.

However, despite the fact that Spotify has music albums organized according to genres, it does not always specify the specific genres of each song. This can make it difficult for users to create playlists or albums based on specific genres, as they may not have all of the information they need to do so accurately.

To address this issue, some music streaming platforms have implemented machine learning models that can predict the genre of a song based on its characteristics and features. These models can help users to more easily find and discover new music that fits their preferences, and can make it easier for users to create playlists and albums based on specific genres.

In this project, we approached the task of predicting the music genre from a different perspective than many other projects in the literature. Most previous projects have focused on selecting only one genre label for each song, and then making a prediction based on that label. However, songs often have multiple labels, and this approach can lead to a loss of important information.

To address this issue, we decided to work with multiple labels for each song in our dataset. We tried to predict one of these labels and counted it as a true positive in the confusion matrix if one of our predictions was correct. This approach allowed us to more accurately capture the diverse range of genres that each song belongs to, and to build more reliable and accurate machine learning models for predicting the genres of new pieces of music.

Overall, our aim was to find the best prediction for the genre of a given piece of music, using a wide range of features and a variety of machine learning techniques. By taking this approach, we hoped to be able to more accurately and comprehensively classify the music in our dataset, and to build more effective models for predicting the genres of new pieces of music.

II. Related works

Most previous genre prediction projects only predict one label for a given piece of music. If the music has multiple labels, the project will select one of those labels to predict. However, in our project, we predict two labels for each piece of music. If one of the predicted labels matches one of the actual labels of the song, we consider that a true positive in our confusion matrix. In a different research paper published in the Transactions of the International Society of Music Information Retrieval, the authors investigate new techniques for generating features for genre classification from song lyrics. These techniques involve using advanced methods to extract information about rhyme scheme, parts of speech, and punctuation in the lyrics. The authors found that these features were more effective than simply using a bag of words to classify the genre. We were unable to find any significant research on using album artwork for image classification in genre prediction.

III. Preliminary Experiments

A. Gathering Data

The first step is to collect data. We collect data with "Spotify API". We take 34,246 tracks of music from lots of genres. Also, the data which is collected has some features.

Data features has include : artistname, id, trackname, danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence, tempo, artistpop, genres, trackpop, genreslist, subjectivity, and polarity.

B. DataSet

First step is collect data. We collected data with ”Spotify API”. We took 34,246 tracks of music lots of genres. Also the data which is collected has features. Data features includes:

Data features
artistname
id
trackname
danceability
energy
key
loudness
mode
speechiness
acousticness
instrumentalness
liveness
valence
tempo
artistpop
genres
trackpop
genreslist
subjectivity
polarity

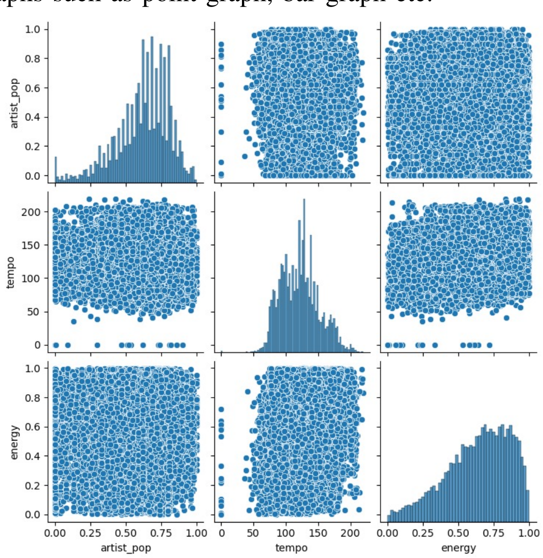
C. Data Visualization

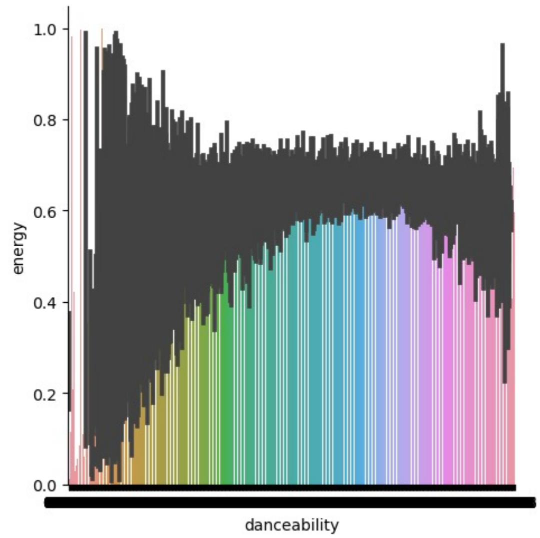
Data is visualized in various ways.

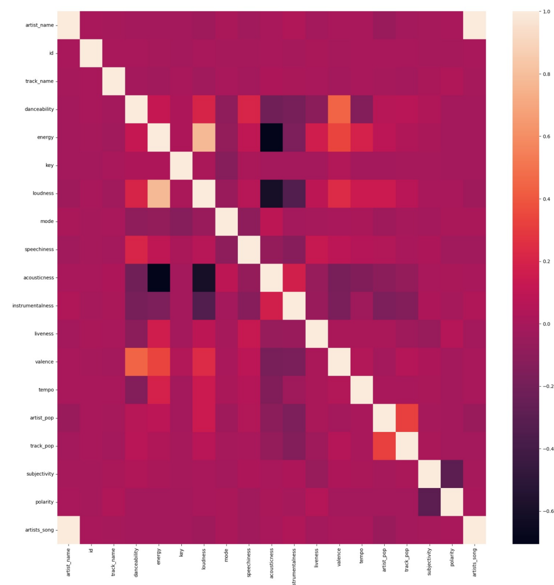
Visualization was done using seaborn vs matplotlib libraries.

The visualization was done by determining the appropriate chart type after the target attributes were determined.

Visualizations (figure 1-2-3-4) were made using various graphs such as point graph, bar graph etc.







D. Data Preprocessing

The data was carefully edited. Trainings will be made on these edited data. With the data preprocessing step, necessary normalization feature transformations were made. Overfitting data and useless data were removed from our dataset. Discarded for not using id, trackname, and genreslist columns.

```
RangeIndex: 34247 entries, 0 to 34246
Data columns (total 17 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   artist_name       34247 non-null  float64
 1   danceability      34247 non-null  float64
 2   energy            34247 non-null  float64
 3   key               34247 non-null  float64
 4   loudness          34247 non-null  float64
 5   mode              34247 non-null  int64
 6   speechiness       34247 non-null  float64
 7   acousticness      34247 non-null  float64
 8   instrumentalness  34247 non-null  float64
 9   liveness          34247 non-null  float64
 10  valence           34247 non-null  float64
 11  tempo             34247 non-null  float64
 12  artist_pop        34247 non-null  float64
 13  genres            34247 non-null  object
 14  track_pop         34247 non-null  float64
 15  subjectivity      34247 non-null  float64
 16  polarity          34247 non-null  float64
dtypes: float64(15), int64(1), object(1)
```

In this project, we used multiple labels for each song in order to more accurately capture the diverse range of genres that each song belongs to. This was important because songs often have multiple labels, and using only one label to classify a song could result in a loss of information.

To collect this data, we applied some operations to initialize the minimum values from zero and decrease the values. This helped us to more accurately capture the genres that each song belonged to, and allowed us to identify the main genres for each song.

To determine the main genres, we used a threshold of 3000. If a genre had a count greater than 3000, we considered it to be a main genre. This allowed us to identify the genres that were most prominent for each song and to use this information to build more accurate and reliable machine learning models for predicting the genres of new pieces of music. Overall, this approach helped us to more accurately and comprehensively classify the music in our dataset.

To identify the main genres of each song in our dataset, we used a loop to iterate through the list of genres and examine the counts of each genre. This allowed us to identify the genres that were most prominent for each song, and to use this information to build more accurate and reliable machine learning models for predicting the genres of new pieces of music.

To do this, we first converted the list of genres into an array and then used a loop to iterate through the array. We examined each label and determined the main genres based on the counts. The main genres that we identified were: contemporary, country, dance, folk, hip hop, house, indie, metal, modern, pop, rap, rock, r&b, soul, and trap. These were determined based on a list of common music genres and were chosen because they represented a wide range of styles and genres within the music industry.

Overall, this approach allowed us to more accurately classify the music in our dataset and to build more effective machine learning models for predicting the genres of new pieces of music.

Genres and Counts
contemporary: 4164
country: 8126
dance: 6327
folk: 3302
hip hop: 10462
house: 3156
indie: 9268
metal: 3809
modern: 4296
pop: 33618
rap: 15151
rock: 22340
r&b: 3466
soul: 2937
trap: 3410
other: not belonging to any genre

In this project, we have 16 main genres in total.

The data were appropriately divided into training, validation, and testing parts, and cross-validation was applied according to the appropriate k value. Random state is selected to be 12.

E. Machine Learning Methods

In our study, we used 29 different machine learning classifiers to identify and analyze the features that are unique to different music genres. These classifiers included popular methods such as Random Forest, Support Vector Machines (SVM), and Logistic Regression.

To train our models, we used a variety of machine learning algorithms and techniques to extract and analyze the features that are characteristic of different music genres. We evaluated the performance of these models using a range of metrics and approaches, and we analyzed the results to gain insights into the strengths and limitations of each method.

Overall, our goal was to identify and understand the specific features that are unique to different music genres, and to use this information to build accurate and reliable machine learning models for predicting the genre of a given piece of music. Through our analysis, we hoped to gain a deeper understanding of the characteristics of different music genres and the factors that contribute to their distinctiveness.

In order to assess the performance of our machine learning models, we used the lazy predict method to build and test a large number of basic models with minimal coding and without any parameter tuning. This allowed us to quickly and easily evaluate the performance of each model and compare their results.

We obtained training results for all 29 models using this method, and among these models, the most commonly used were Random Forest, Logistic Regression, and SVM. These models are popular because they are relatively simple to implement and can achieve good results on a wide range of tasks.

In addition to examining the overall performance of these models, we also looked at their results in more detail to gain a deeper understanding of their strengths and limitations. This included analyzing the specific features that were most important for predicting the genre of a given piece of music and identifying any patterns or trends in the data. By doing this, we hoped to gain insights that would help us improve the accuracy and reliability of our models.

1. Logistic Regression (LR): This linear classifier is generally used for binary classification tasks. For this multi-class classification task, the LR is implemented as a one-vs-rest method. That is, 16 separate binary classifiers are trained. During test time, the class with the highest probability among the 16 classifiers is chosen as the predicted class.

2. Random Forest (RF): Random Forest is an ensemble learning that combines the prediction from a pre-specified number of decision trees. It works on the integration of two main principles: Each decision tree is trained with only a subset of the training samples which is known as bootstrap aggregation, Each decision tree is required to make its prediction using only a random subset of the features.

3. Support Vector Machines (SVM): SVMs transform the original input data into a high dimensional space using a kernel trick. The transformed data can be linearly separated using a hyperplane.

It is not uncommon for machine learning models to have relatively low prediction accuracy, especially when working with complex or noisy data. There are many factors that can affect the performance of a machine learning model, including the quality and quantity of the training data, the choice of features and algorithms, and the complexity of the task.

In this case, it seems that the Random Forest classifier had the highest prediction accuracy among the models that were tested. While this accuracy may still be considered low, it is likely that other machine learning models or different approaches would yield different results. It is important to carefully evaluate the performance of a machine learning model and consider the context in which it will be used before making any decisions or drawing conclusions based on its predictions.

Model	Accuracy	Balanced Accuracy	ROC AUC	F1 Score	Time Taken
NearestCentroid	0.14	0.26	None	0.13	0.04
QuadraticDiscriminantAnalysis	0.28	0.22	None	0.29	0.13
GaussianNB	0.22	0.21	None	0.23	0.07
AdaBoostClassifier	0.32	0.20	None	0.30	1.77
LGBMClassifier	0.41	0.19	None	0.38	4.00
XGBClassifier	0.40	0.18	None	0.37	15.41
LabelPropagation	0.28	0.16	None	0.28	16.50
LabelSpreading	0.28	0.16	None	0.28	22.08
LinearDiscriminantAnalysis	0.38	0.16	None	0.34	0.12
BaggingClassifier	0.37	0.16	None	0.34	2.28
RandomForestClassifier	0.42	0.16	None	0.36	7.69
BernoulliNB	0.32	0.15	None	0.29	0.05
ExtraTreesClassifier	0.39	0.15	None	0.34	5.26
LogisticRegression	0.39	0.14	None	0.34	1.38
KNeighborsClassifier	0.33	0.14	None	0.31	1.43
DecisionTreeClassifier	0.26	0.14	None	0.27	0.47
SVC	0.40	0.13	None	0.34	31.22
ExtraTreeClassifier	0.24	0.13	None	0.24	0.07
CalibratedClassifierCV	0.38	0.12	None	0.32	61.44
LinearSVC	0.38	0.11	None	0.31	16.78
PassiveAggressiveClassifier	0.28	0.11	None	0.24	0.29
RidgeClassifier	0.37	0.10	None	0.30	0.07
RidgeClassifierCV	0.37	0.10	None	0.30	0.12
Perceptron	0.23	0.10	None	0.23	0.19
SGDClassifier	0.31	0.09	None	0.25	0.67

F. Neural Network (Multi Layer Perceptron)

A multilayer perceptron (MLP) is a fully connected class of feedforward artificial neural network (ANN). It is a class in the scikitlearn library that implements a multi-layer perceptron (MLP) for classification. Strictly to refer to networks composed of multiple layers of perceptrons. An MLP consists of at least three layers of nodes: an input layer, a hidden layer and an output layer. Except for the input nodes, each node is a neuron that uses a nonlinear activation function.

We change the parameters to the find the best results.

Learning rate,

hidden layer number,

hidden layer size,

gamma,

batch size,

shuffle,

early stopping number

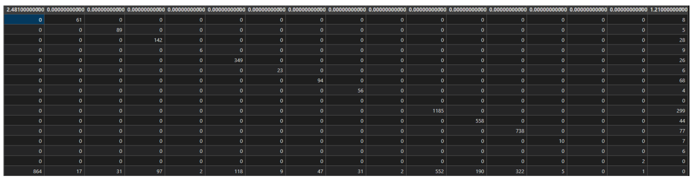
For hidden layer sizes we used 1 and 2 hidden layers, to see the effect to our dataset. We try 3 hidden layer but there was not much change. For this reason, we used 2 hidden layers. The best hidden layer size is 128 / 64. Learning rate 0.1 , epoch number 1000, early stopping number 10, gamma 0.8, shuffle false are best parameters.

See Table 1 for the best parameters.

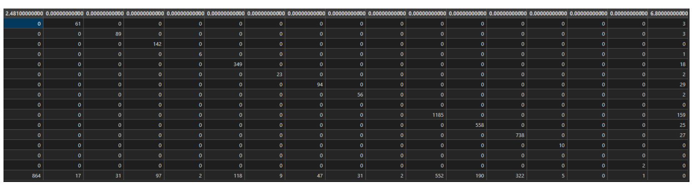
Parameters	Using Value
Learning Rate	0.01
Hidden Layer	2
Hidden Layer Size	128, 64
Gamma	0.8
Batch size	1
Activation	relu
Shuffle	false
Early Stopping Number	10

a) Confusion Matrix: Confusion matrix is a tabular representation which enables us to further understand the strengths and weaknesses of our model. Element aij in the matrix refers to the number of test instances of class i that the model predicted as class j. Diagonal elements aii corresponds to the correct predictions.

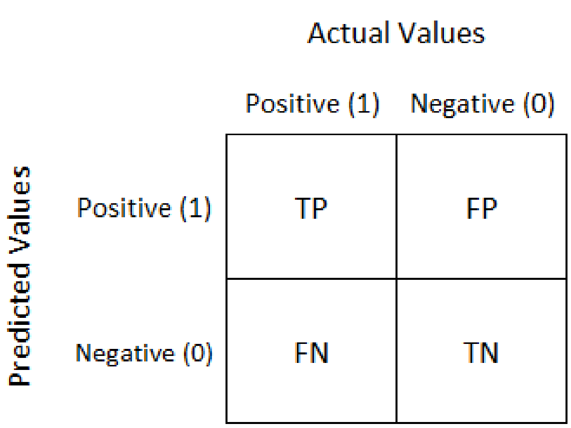
Confusion Matrix



Confusion Matrix 2 Predict



b) Accuracy: Refers to the percentage of correctly classified test samples. Calculation is made according to the hit-miss numbers of the genres. Accuracy results are calculated. Hit-miss information of predictions is kept for each genre with Confusion Matrix.



- About a single accuracy each genres:

For each Genre
72.28485015122354
75.0
67.8082191780822
66.43835616438356
80.23255813953489
75.32956685499059
76.36363636363636
76.36363636363636
66.33663366336634
90.9090909090909
67.1280276816609
75.05827505827506
69.96779388083736
58.53658536585366
69.44444444444444
83.33333333333333

- About the two accuracy each genres:

For each Genre
73.0
71.71717171717172
67.4074074074074
65.47085201793722
64.0
74.89711934156378
70.96774193548387
62.27272727272727
65.21739130434783
66.66666666666667
64.57725947521865
75.09627727856225
70.31963470319634
61.29032258064516
62.5

G. Discussion

In this project, we researched what we can do to predict the genre of a given music. The projects in the literature were examined and researches were made to do something that has not been done until now. And unlike the projects in the literature, we tried to predict the music genre with the multi-label perceptron. We made 2 guesses as a piece of music can have more than one genre.

Data was organized with data visualization and preprocessing steps. Machine learning methods were used, but despite increasing the data, the result was not above 42%.

For this reason, a multi-label sensor was used. Also, the music genre data contained a lot of data. These data are reduced to a list of the main species. Those that contain the main species are added to the list. In this way, the number of music genres is reduced. Genres below 3000 music genres and songs not found in the main genres are placed on the 'other' list. With this preprocessing and the model used, the results were improved. Hit-miss information of predictions is kept for each genre with Confusion Matrix. An error matrix is created for each type and accuracy is calculated. Estimates are made. As a result, it predicts 65.9% with 82.28% accuracy. If 2 predictions are made for a music, the accuracy is 82.28% to prediction 68.8

H. Conclusion

In this study, we tried to predict the music genre of a song using Spotify API data.

We tried two different approaches to solving this problem.

These approaches are:

1) Using classification methods with machine learning,

The results obtained as a result of machine learning were Random Forest Classification with a maximum of 42%.

2) Multi-Layer Perceptron

The learning results using Multi-Layer Perceptron were compared in two ways.

- About the one predictions:

For each genres 71.08797197353056%

guessing correctly 65.9%

accuracy 82.28%

average accuracy of all music genres separately is 70.8%

- About the two predictions:

for each genres 90.9090909090909%

guessing correctly 68.8%

accuracy 82.28%

average accuracy of all music genres separately is 70.01%

The success of our machine learning project can be evaluated based on the results that it produces. In an effort to make our project more easily accessible and user-friendly, we utilized the Streamlit library for deployment. This library allows code to be run either locally or on a website and provides several functions for designing user interfaces and interacting with users. Additionally, it facilitates the deployment of projects through the use of Python files, model pickles, and a requirements.txt file. To deploy our project, we first uploaded these files to Github and then used the Streamlit website as the host.

On our website, users can input various parameters and the multi layer perceptron (MLP) will return predicted labels based on those inputs. The website is dynamic and will change based on the user's input. If you are interested in seeing the code and updates for our project, you can follow the commits on our Github page. https://github.com/HuseyinEmreAksoy/SpotifySongPrediction

References

[1] Q. H. Nguyen et al., "Music Genre Classification using Residual Attention Network," 2019 International Conference on System Science and Engineering (ICSSE), 2019, pp. 115-119, doi: 10.1109/ICSSE.2019.8823100.

[2] Ghildiyal and S. Sharma, "Music Genre Classification Using Data Filtering Algorithm: An Artificial Intelligence Approach," 2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA), 2021, pp. 922-926, doi: 10.1109/ICIRCA51532.2021.9544592.

[3] Bahuleyan Hareesh, "Music Genre Classification using Machine Learning Techniques".

[4] Irfan Siddavatam, Ashwini Dalvi, Dipen Gupta, Zaid Farooqui, Mihir Chouhan "Multi Genre Music Classification and Conversion System".

[5] Li Guo(liguo94), Zhiwei Gu(zhiweig), Tianchi Liu(kitliu5), "Music Genre Classification via Machine Learning".

[6] Gershon, Ariel Amar Hanan Benzion Elad, Ruth Taboada. "AI Genre Classification by Lyrics.".

[7] https://ieeexplore.ieee.org/document/8554816

[8] FEATURE EXTRACTION USING MFCC Shikha Gupta1 , Jafreezal Jaafar2 , Wan Fatimah wan Ahmad3 and Arpit Bansal4 Signal & Image Processing : An International Journal (SIPIJ) Vol.4, No.4, August 2013.

[9] Y. Huang and L. Li, "Naive Bayes classification algorithm based on small sample set," 2011 IEEE International Conference on Cloud Computing and Intelligence Systems, 2011, pp. 34-39, doi: 10.1109/CCIS.2011.6045027.

[10] B. Logan, Mel Frequency Cepstral Coefficients for Music Modeling, Transactions of the International Society of Music Information Retrieval. Ubiquity Press, 2000.