Predicting Song Genre With Audio Features Using Neural Networks

Amy Lee, Rohan Mathur, Anika Vyas, Ethan Yoon

I. Introduction

This study investigates how powerful a song's audio features are in predicting its genre. Music genres are categories or styles of music which share unique features, such as instruments, rhythm, melody, and lyrics. Those features help distinguish each genre from others. A recommender system suggests items based on a user's behavior and preferences. They are widely used in industries such as e-commerce, entertainment, and social media to increase engagement and sales. The genre of a song is an important factor in a music recommender system, as it can be used to suggest similar songs that align with a user's musical tastes and preferences. However, music genre classification is a task that is often subjective and ambiguous. By utilizing machine learning, we aim to reduce subjectivity and provide a more standardized and consistent approach to genre classification. Doing so can help make sense of the vast amount of music that is out there. We will be able to unlock new insights and applications in the music industry and provide better recommendations and experiences for music listeners.

II. Current literature

In the study "Classification of Music Genres using Feature Selection and Hyperparameter Tuning," the authors aimed to identify the optimal classification method for genre classification using the Kaggle Spotify dataset. They assessed several machine learning algorithms, including logistic regression, kNN, SVM, XGBoost, and Random Forest. The resulting test accuracies varied between 51.19% and 99.6%, with Random Forest achieving the highest accuracy. Additionally, feature selection was performed to determine the most influential features for genre classification, revealing popularity, acousticness, and danceability as significant predictors.

In the study titled "Predicting Musical Genres using Deep Learning and Ensembling," the author focused on leveraging deep learning techniques to develop an Automatic Musical Genre Classification (AMGC) model. The dataset used for this study was the million song dataset (MSD). The author constructed three separate models for genre prediction. The first model was trained on lyrics, utilizing text-based analysis of song lyrics to predict genres. The second model was trained on cover art, using image-based analysis of album cover art to predict genres. The third model incorporated a range of musical features, such as tempo, timbre, and pitch, to predict genres.

After training these individual models, the author combined their predictions to create a final prediction. This ensemble approach aimed to leverage the strengths of each model and improve overall genre classification accuracy. By utilizing deep learning techniques and ensembling the predictions from different models trained on lyrics, cover art, and musical

features, the study aimed to develop an effective AMGC model using the MSD dataset. The combination of multiple modalities was intended to enhance the accuracy and robustness of genre predictions.

III. The Data

A. Dataset Description

For our investigation, we are using a Spotify Tracks database, last edited in 2019 and downloaded via the Spotify API. The database uses 15 classifiers to distinguish the songs. The user that uploaded the data onto Kaggle sourced it directly from the Spotify API, so the data can be considered reliable. Spotify published a description of the audio features provided via its API. We have included a brief description of all the features in the following table.

Table I: Description of all the features of the Spotify dataframe

Audio Feature	**Description**
acousticness	Confidence measure ranging 0.0 to 1.0 of whether the song is acoustic
danceability	Measure ranging 0.0 to 1.0 of danceability (based on the tempo, rhythm stability, beat strength, and overall regularity of the song)
duration_ms	Duration of the track in milliseconds
energy	Measure ranging 0.0 to 1.0 of intensity and activity (features of energy include dynamic range, perceived loudness, timbre, onset rate, and general entropy)
instrumentalness	Measure ranging 0.0 to 1.0 of the instrumentalness of a song using its vocal content; scores greater than 0.5 are considered instrumental
key	Track key (A, A#, B, C, C#, D, D#, E, F, F#, G, G#)
liveness	Measure ranging 0.0 to 1.0 that detects the presence of an audience in the audio, value above 0.8 provides strong likelihood that the song is live
loudness	Overall loudness of a track in decibels (dB), values typically range between -60 and 0 dB
speechiness	Measure from, 0.0 to 1.0 of the presence of spoken words in a track
tempo	Overall estimated tempo of the track in beats per minute (bpm)
valence	Measure from 0.0 to 1.0; describes the track's musical positiveness (closer to 0 suggests the song sounds more negative)

B. Data Exploration

To better understand the data, we carried out some data exploration. First, we plotted a histogram for each of our twelve features (Figure I) and a correlation hot map (Figure II). Some important observations can be made from these figures.

From the histograms, we can see how “instrumentalness” and “speechiness” have slight bimodal distributions. The variable "duration_ms" has a much larger scale than many of the other numerical variables in the dataset, being approximately 400 thousand times larger. Most of the data for duration_ms is concentrated around 250 thousand ms. Moreover, the x-axis extends all the way to 60 million, suggesting that outliers may exist.

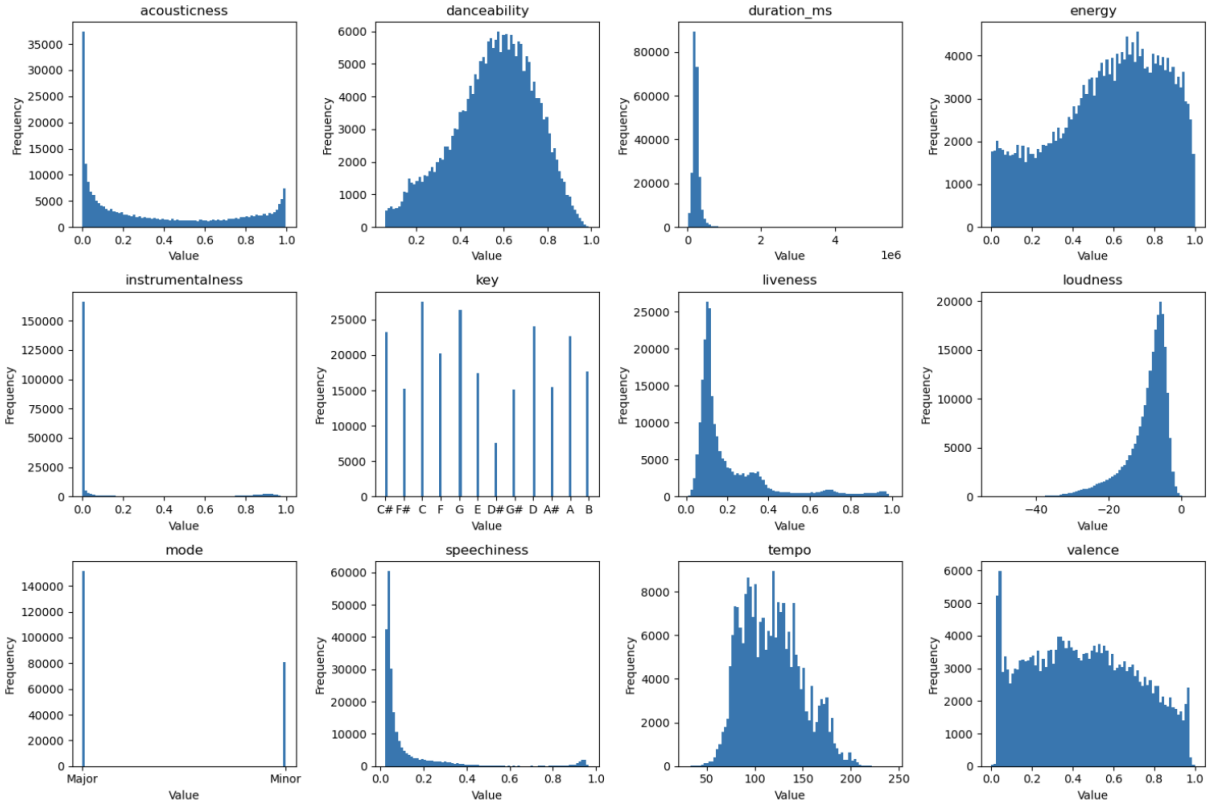

Figure I: Histogram of each of the 12 features, unscaled and unfiltered

Secondly, we calculated the correlation between all of the features and genres of the dataset as seen in Figure II. Because correlation can only be calculated using numerical variables, we had to encode the “key” feature using circle of fifths encoding and one-hot encoded “mode” (more details in section III.C). At first glance, the graph has an overwhelming number of correlations that are close to 0. The genres seem to have a low correlation with the features being used to predict it; however, all the predictors have a relatively higher correlation amongst themselves. This is worrying since we will be using the features to predict the genre and if they have a low correlation, it would imply that our model may not have the most accurate results.

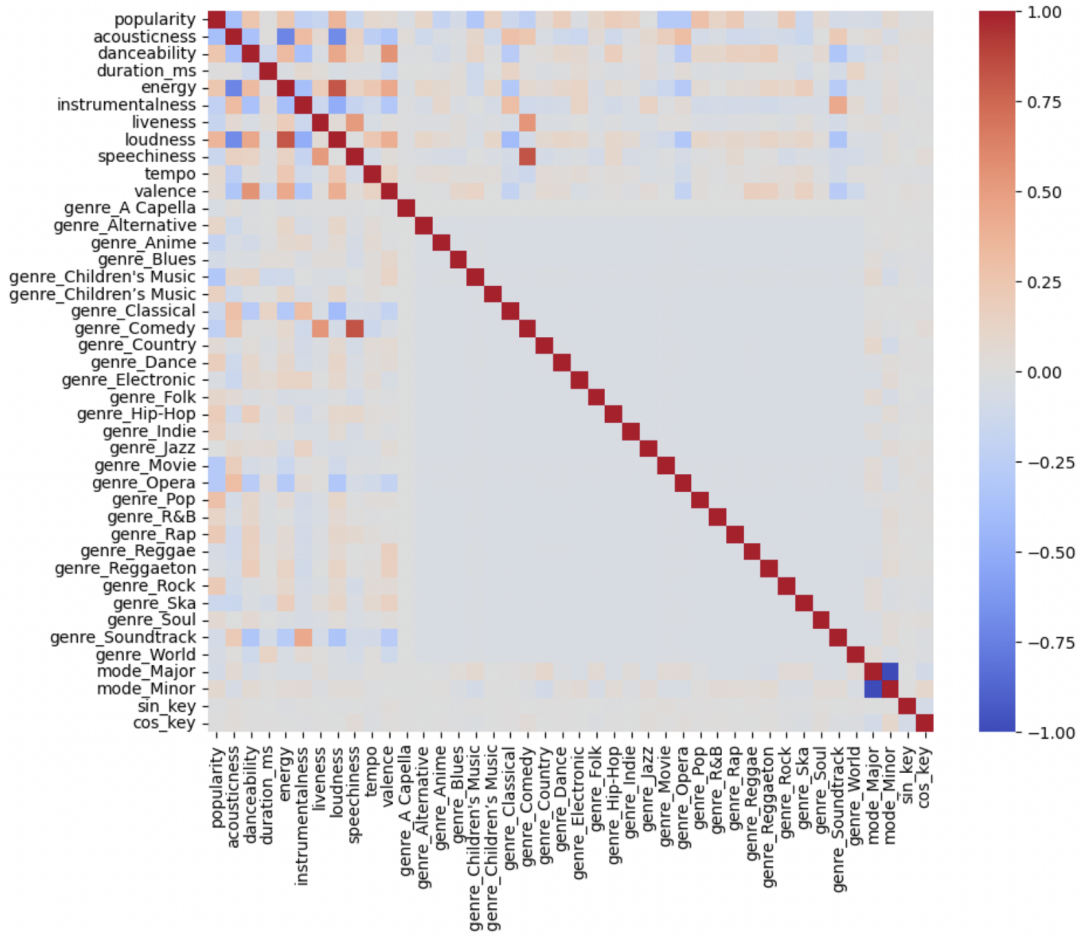

Figure II: Heatmap of the Correlation Between all Features and Genres

C. Pre-Processing The Data

We examined each of the twelve features by genre to determine which data points to filter out, as certain genres were expected to have relatively more extreme feature values than others. After examining the features by genre, we determined that there are actually no outliers to filter out. We also noticed that some songs were included more than once, that is some songs were associated with more than one genre. This could reduce our accuracy as songs with the exact same feature specifications would be plotted to multiple different classes, confusing our model. As a result, we removed all of those duplicates.

We split our data into 80/20 train and testing sets ensuring that each genre was proportionally represented in both datasets. To ensure that each variable is given equal weight in the analysis, we scaled the features using robust scaling. This method uses the median and IQR, which will help account for relatively extreme feature values per genre. We also encoded the categorical variable. The testing and training data were scaled based on the training data.

One hot encoding was performed on the “genre” and “mode” features. Moreover, circle of fifths encoding was performed on “key” to preserve harmonic distances and make sure they have the right symmetry. The sin and cos components that are used to represent the position of a musical key on the circle of fifths are defined as follows,

$$sin\_key = sin(2\pi\frac{key}{12})$$
$$cos\_key = cos(2\pi\frac{key}{12})$$

IV. Model Implementation

We decided to use a neural network model since there is a lot of non-linearity and depth of regression with these subjective-objective facts that we are using as features. We limited our initial models to three hidden layers. We performed cross-validation with the accuracy score as our evaluation metric to select the number of nodes for each hidden layer, the activation function, and hyperparameter alpha. Alpha controls the L2 penalty term added to the weights of the neural network. We tested with (100, 100, 100), (50, 100, 50), (100, 100, 100), (50, 100, 100), (100, 150, 100), and (100,) for the size of each hidden layer; tanh and relu for the activation function; 0.001, and 0.5 for alpha. The best model had hidden layer size (50, 100, 100), activation function tanh, and an alpha of 0.05 (Model A). With this model, we produced an accuracy score of 34%. The log loss (2.72) and AUC (59%) both indicate that the model is poor, but performing slightly better than chance.

Unsatisfied with this accuracy score, we decided to experiment with our parameters to improve its predictive capabilities. Model A only used one type of activation function, so we tried a more complicated model to try and get our loss to converge to a more acceptable level (~1). We ran a 100-epoch neural network with three hidden layers on our training set (Model B). Each hidden layer uses a relu, tanh, and softmax, respectively. We chose relu for the sake of time and complexity, as it is the fastest to compute and take derivatives of. Tanh was also the best activation function from our cross-validation search. In addition, it is standard to use softmax as the last activation function to turn your results into vectors. While we did see an increase in accuracy, loss, and AUC, the model was still not very successful in predicting genres. A summary of performance metrics for our models is in Table I of section V.

V. Results

	Accuracy	**Log Loss**	**AUC**	**Precision**
Model A	34%	2.72	59%	0.49
Model B	37.9%	2.07	89.3%	0.75

Table II: Performance Metrics for Each Model

In order to get a better idea of the performance of the models for each of the genres, rather than as a whole, we made a confusion matrix that measures the number of true positive outputs of the model (Figure IV and Figure V). The diagonal across the two figures shows us the number of genres that were correctly labeled. The remaining grid squares represent the number of misclassifications across all the genres.

Figure IV indicates that the relationship between predicted and true values in model A is not especially strong. The diagonal alignment in the graph is not as clear as expected in an accurate model, showing significant deviation. However, the model performed relatively well in predicting the comedy and soundtrack genres, correctly classifying close to 1600 and 1200 tracks, respectively. In contrast, there were noticeable prediction errors in most other categories. Specifically, a distinct row of dark-colored squares along the y-axis above the A-capella genre indicates a tendency of model A to misclassify data to this genre. This tendency seems to have been the strongest in the case of soul, rock, jazz, country, and blues music.

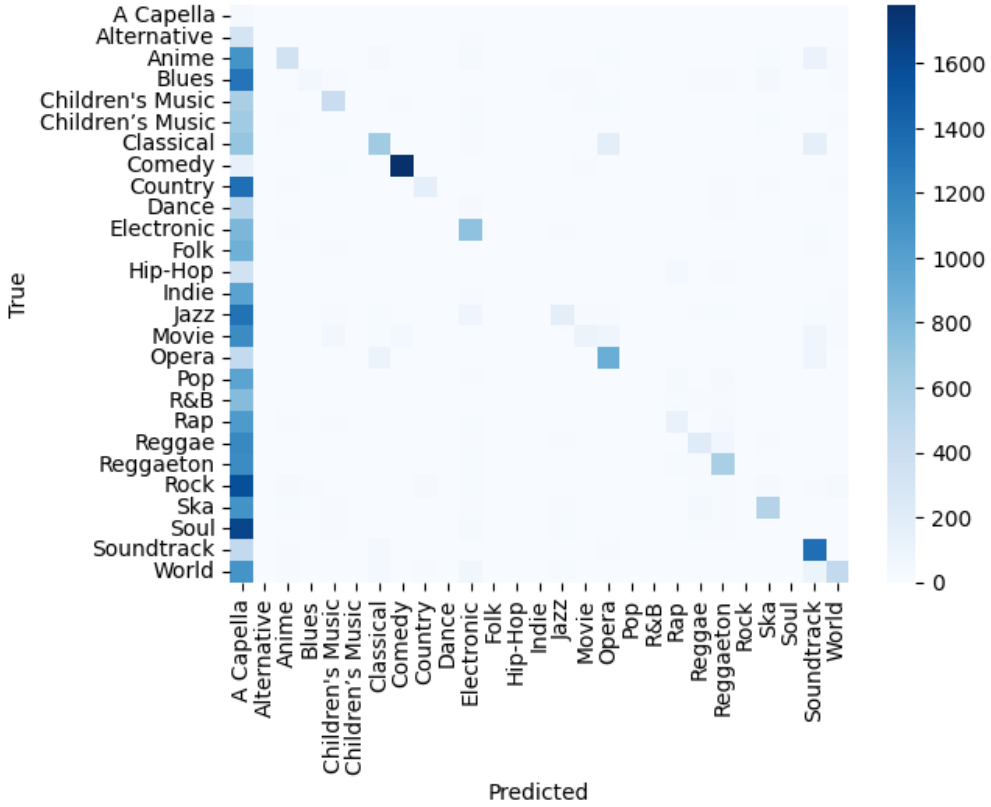

Figure IV: Confusion Matrix of predicted vs true labels for Model A

Looking at Figure V below, we can discern that Model B correctly classified ‘Soundtrack’ and ‘Comedy’ most of the time. Genres like ‘Opera’, ‘Reggaeton’, ‘World’, and

‘Ska’ also had a relatively high true positive rate, when compared to some of the other genres like ‘Rock’, ‘Hip-Hop’ and ‘Folk’ which had a true positive number under 400. This might also be due to the testing dataset not having a balanced count. To get a better understanding of the distribution of songs across all the genres, look at the appendix (Figure VI).

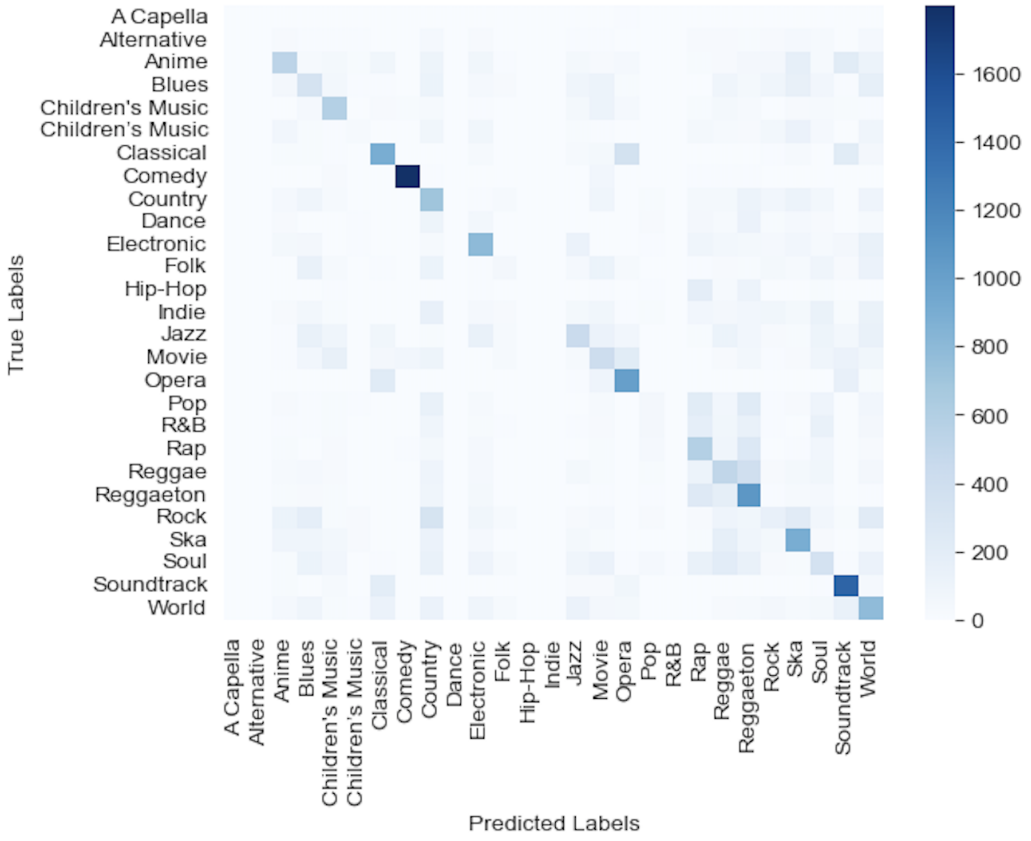

Figure V: Confusion Matrix of predicted vs true labels for Model B

Comparing Figure IV to Figure V suggests that Model B performs better than Model A. Model B shows a more pronounced diagonal and lacks the line along the y-axis above the A capella genre, indicating an improvement in misclassification. However, model B does exhibit some random variation in misclassification. It struggles to differentiate between Rap, Reggae, and Reggaeton, as well as Jazz and Movie genres. Similarities include both models correctly classifying comedy and soundtrack genres at a high rate.

VI. Conclusion

At first, it seemed like these more complicated models would be successful in providing us with a high number of true positive results. However, we were surprised to see low accuracy rates across the board for both of the models. While there might have been several reasons behind the low accuracy rates for these models, the low levels of correlation between the genres

and the predictors might have been the most influential reason behind the poor performance and high loss of the model. The low accuracy rates persisted despite the fact that neural networks would be a lot better than traditional models at learning and modeling non-linear and complex relationships. It is also possible that the data frame used isn't best suited to a neural network model, resulting in the low accuracy rates. Nonetheless, it is important to note that both of our models performed around 10x better than random classification which would have been about a 3.57% accuracy.

VII. Contributions

All group members contributed to the project and touched every part of it. Amy performed data exploration by creating histograms, did the data pre-processing, and trained/tested possible models. Rohan researched and wrote about current literature. Anika described the dataset, performed more data exploration by creating the correlation heatmap, and helped with investigating what models to use. Ethan worked on the model implementation by training, testing, and improving models; he also created the heatmaps for each of the models. All group members contributed to the write-up and analysis of our results.

VIII. Appendix

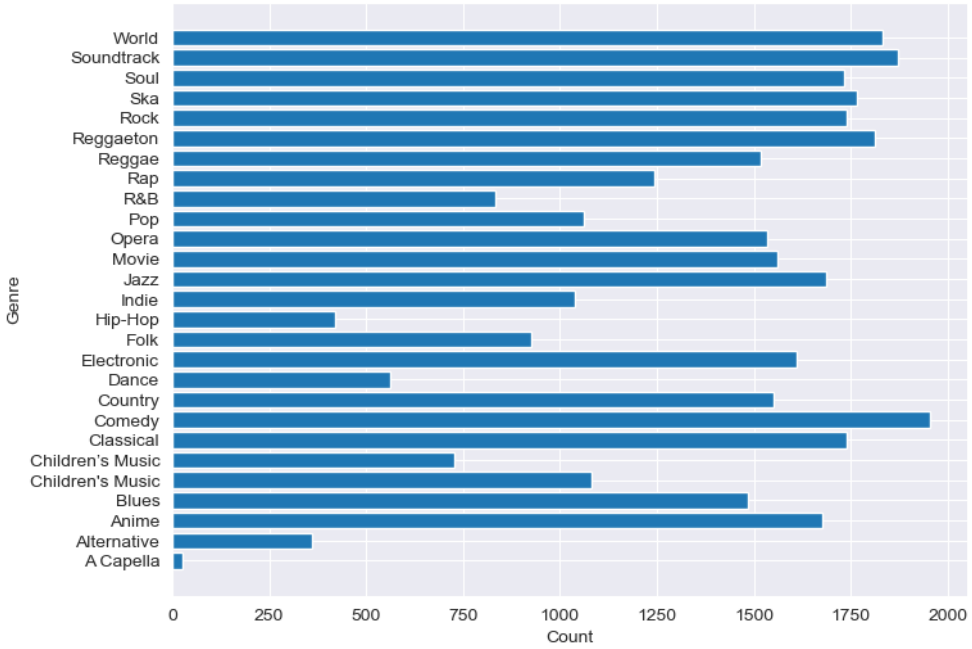

Figure VI: Barplot of the Testing Dataset Genre Count

Works Cited

Sang, A. M. “Predicting Musical Genres Using Deep Learning and Ensembling.” *UCLA Electronic Theses and Dissertations*, 2020.

Singhal, Rahul, et al. “Classification of Music Genres Using Feature Selection and Hyperparameter Tuning.” *September 2022*, vol. 4, no. 3, 2022, pp. 167–178, https://doi.org/10.36548/jaicn.2022.3.003.