# MUSIC RECOMMENDATION SYSTEM

*by* Mukul Jain

**Submission date:** 14-Apr-2024 04:07PM (UTC+0530)
**Submission ID:** 2348995890
**File name:** Minor_Project1FINAL_1_1.doc (2.01M)
**Word count:** 4958
**Character count:** 28121

A PROJECT REPORT

on

# "MUSIC RECOMMENDATION SYSTEM"

Submitted to

**KIIT Deemed to be University**

In Partial Fulfilment of the Requirement for the Award of

**BACHELOR'S DEGREE IN INFORMATION TECHNOLOGY**

BY


| NAME | ROLL NO |
|---|---|
| Mukul Jain | 2129078 |
| Alkesh Singh | 2129021 |
| Ashutosh Dubey | 2129128 |
| Pradyumn Mukhopadhyay | 2129082 |

UNDER THE GUIDANCE OF

Prof. Sourav Kumar Giri

SCHOOL OF COMPUTER ENGINEERING

KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY

BHUBANESWAR, ODISHA - 751024


APRIL 2024

A PROJECT REPORT

on

"MUSIC RECOMMENDATION SYSTEM"

Submitted to

KIIT Deemed to be University

In Partial Fulfilment of the Requirement for the Award of

BACHELOR'S DEGREE IN
INFORMATION TECHNOLOGY

BY


| NAME | ROLL NO |
|---|---|
| Mukul Jain | 2129078 |
| Alkesh Singh | 2129021 |
| Ashutosh Dubey | 2129128 |
| Pradyumn Mukhopadhyay | 2129082 |

2

UNDER THE GUIDANCE OF

**Prof. Sourav Kumar Giri**




SCHOOL OF COMPUTER ENGINEERING

KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY

BHUBANESWAE, ODISHA -751024


APRIL 2024

KIIT Deemed to be University
School of Computer Engineering
Bhubaneswar, ODISHA 751024

# CERTIFICATE

This is certify that the project entitled

"MUSIC RECOMMENDATION SYSTEM"

submitted by

| NAME | ROLL NO |
|---|---|
| Mukul Jain | 2129078 |
| Alkesh Singh | 2129021 |
| Ashutosh Dubey | 2129128 |
| Pradyumn Mukhopadhyay | 2129082 |

2
is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Science & Engineering OR Information Technology) at KIIT Deemed to be university, Bhubaneswar. This work is done during year 2022-2023, under our guidance.

Date: / /

**Prof. Sourav Kumar Giri**
Project Guide

# Acknowledgements

We are profoundly grateful to **PROF. SOURAV KKUMAR GIRI** of **SCHOOL OF COMPUTER SCIENCE AND ENGINEERING** for his expert guidance and continuous encouragement throughout to see that this project rights its target since its commencement to its completion. .....

MUKUL JAIN

ASHUTOSH DUBEY

ALKESH SINGH

PRADYUMN MUKHOPADHYAY

# ABSTRACT

Creating a recommendation system for the automated playlist continuation duty is the aim. A recommender system should be able to provide a list of suggested songs that may be added to a playlist, thus "continuing" it, given a set of playlist attributes. The "Million Playlist Dataset" supplied by Spotify will be utilized as the dataset for our model's training. Playlist continuation is achieved by accurately predicting and recommending songs to the appropriate playlists. Finding different ways to suggest the next song to the user or playlist is the key challenge at hand. There is much debate about whether it should be determined by popularity, performers, genre, language, or music kind.The objective is to develop a recommendation system for the automatic playlist continuation task. Given a set of playlist attributes, a recommender system need to be able to return a list of recommended songs that might be added to a playlist, therefore "continuing" it. We will use Spotify's "Million Playlist Dataset" as the training dataset for our model. The process of correctly anticipating and suggesting music to the right playlists allows for playlist continuance. The main task at hand is coming up with novel ideas for suggesting the next music to the user or playlist. Whether it should be based on popularity, artists, genre, language, or type of music is a topic of intense dispute.

Since there are many methods and approaches from which this can be approached, we moved forward with emotion tracing and prediction. We have a huge dataset and we have classified every song of that dataset with a bunch of emotions which they can possess and marked them. Then we have used machine learning models to accurately classify and predict the songs.

**Keywords:** Music, Recommendation, KNN, Decision Tree, Logistic Regression, Random Forest.

# Contents:-

# Chapter 1

## Introduction

Music recommended systems (MRS) have recently exploded in popularity thanks to music streaming services like Spotify, Pandora and Apple Music.These services hold a very large chunk of customers because of their provided services.Whereas most of the recommended systems have been into existence from quite a few time by now and they are differed from each other in some very significant ways.For e.g. the duration for a song is 3-5 min whereas for a movie it could be for about 2 hours or more than that, the size of the their catalogs contains millio of songs in them,that's why Music Recommended Systems have been requiring different approaches from traditional recommended systems.

This project's goal is to provide automatic playlist continuation based on user's choices and preferences or genres he's listening or usually listens. Furthermore, in this recommendation system the requirement of Only minimal information, such as the playlist's title, the music that are currently in it, and the artists that are linked with those tracks, are needed as input instead of rich and diverse user data.
is more than enough for the system to select the songs accordingly.

Here are some business objectives that would be covered by the project:-
i) Objective of the project is to create a recommended system for automatic playlist continuation from taking the least amount of data by the user(in spotify here).
ii) Given here are some featur, the recommended system in order to "continue" the playlist, systems ought to be able to provide a list of suggested songs that may be added. Next, we must choose how we are going to use this dataset to train our model. analyzing the dataset and determining relationships between the many tracks offered. based on the user's choices, choose the next song for the playlist using a variety of categorization approaches. There might be other strategies, like: Non-Customized Methods: It involves suggesting content that is well-liked throughout the whole system. Uncomplicated and comparatively simpler to execute.

# CHAPTER 2

# LITERATURE REVIEW

Once we used to be very surprised to see how efficient the spotify recommendation system works for users with so much grace. As time went on, we developed more and better recommendation technologies that eventually made it possible for us to locate your listeners and establish a connection between the artist and the listeners—even if you're the rarest or most distinctive kind of musician in the world, doing something that only twenty to twenty-five other people find interesting. This was one of the main inspirations behind our decision to pursue this project, which involved creating a Spotify-like music recommendation engine utilizing a variety of machine learning approaches. The ability of Spotify's music recommendation algorithm to suggest the ideal song, playlist, or even "daily mix" has always been welcomed by us.

Personalized Music Recommendation Systems issues adapting to the individual user or listener's desires, interests, and preferences in different ways so that users don't have to deal with giving too many information too the system for recommending a song or artist or to create an automatic playlist . They're tools for suggesting songs to the users.

Recommendation systems offer personalized, distinctive content and service recommendations to consumers, helping them to manage the issue of information overload. Recently, many techniques have been developed for creating recommendation systems: content-based filtering, hybrid filtering, and collaborative filtering. The most advanced and popular method is collaborative filtering.

Content and collaborative based filtering and association rule mining
It was made with the incentive of improving the hybrid system as a replacement music recommendation system by combining options of primarily based filtering, cooperative filtering, and association rule mining. There were several surveys which show that numerous parameters like content and quality of the songs were reccommended by doing cooperative filtering of ratings by some selected/anonymous userssor customers. The aim of this technique is to reccomend songs to the client that suits their interest. This technique works offline and stores recommendations within the customer's internet profile. It finds out the genre of the song that the customer has listened earlier, like a like sad,rock,hip-hop etc. from the consumer's internet profile.

Collaborative recommendation systems

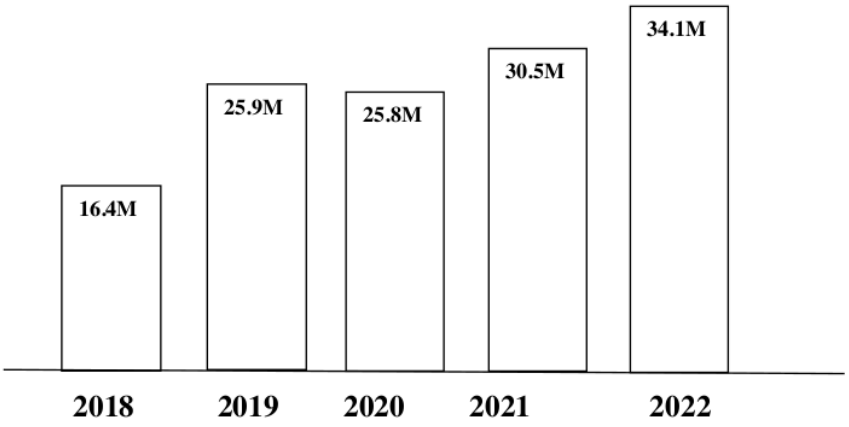
On the website where users begin organizing their music library with tags and also keep track of the music, collaborative filtering finds other users who share the current user's tastes and uses their recommendations on songs or artists that users either upload from their previous playlists or add as favorites.This software creates amazing playlists and filters recommendations based on popularity, genre, and decade. Numerous studies and polls have revealed that collaborative filtering often provides better suggestions than content-based filtering. This is only accurate, though, if use history is accessible, such as song ratings from the past. In the event that this is not the case, it won't prove its accuracy.

.

# Chapter 3

# Problem Statement / Requirement Specifications

According to Luminate, a music data collection company, every day 1,20,000 new songs are uploaded. Thats 1.25 songs every second and 43 million in a year alone. To listen and categorise every song is humanly impossible. Thats why it is a necessity to develop a music recommendation system that can listen, analyse, categorise, generalise and sort music into libraries automatically and suggest suitable songs to the users.




**NEW IRCS CREATED EACH YEAR**

The aim of this project is to see different recommendation approaches which can be used on the available dataset and can be further be suitable for recommending songs beyond the dataset. We have observed distinct machine learning models to validate and verify the system. The model is now trained enough to recommend songs.

## 3.1 Project Planning

The project will follow the following steps:

- Collect data of the songs
- Identify the relevant attributes that may affect the recommendation system
- Train the classification model using the collected data and attributes.
- Test the accuracy of the model using a separate dataset.
- Use the model to predict which song to recommend.

## 3.2 Project Analysis

**Data Gathering -** Data is gathered from the given dataset in CSV format(dataa.csv). The training dataset is used to train the machine learning models while the testing dataset is used to evaluate the performance of the models.

**Data Mining** - The process of extracting patterns, trends, and insights from huge databases is known as data mining. It comprises applying a variety of methods, including statistical analysis, machine learning, and pattern recognition, to extract actionable insights and information from unprocessed data.

**Data Pre-processing -** Data preparation is the process of transforming raw data into a form that can be understood. This stage of the data mining process is also very important as we cannot work with raw data.
Data Pre-processing have three main functions.

- **Data Cleaning -** Finding and addressing outliers, inconsistent data, and missing values in the dataset to guarantee its accuracy and dependability.
- **Data Normalisation** - Also known as Data Standardization ensuring that each feature contributes equally to the model by scaling it to a comparable range.
- **Data Encoding -** use methods like one-hot encoding or label encoding to convert category values into numerical representation

## 3.3 Dataset Description

The dataset used in this project is in “dataa.csv” file. It is taken from the spotify recommendation system. This dataset contains metadata and attributes of 196 songs. This dataset has been through data mining techniques to make it suitable for this model.

Dataset Attributes -

*danceability* of the song
*energy* of the song
*key* of the song
*loudness* of the song
*mode* of the song
*speechiness* of the song
*acousticness* of the song
*instrumentalist* of the song
*liveliness* of the song
*valence* of the song
*tempo* of the song
*duration_ms* of the song
*time_signature* of the song
*liked* song

| danceability | energy | key | loudness | mode | speechiness | acousticness | instrumentalness | liveness | valence | tempo | duration_ms | time_signature | liked |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0.803 | 0.624 | 7 | -6.764 | 0 | 0.0477 | 0.451 | 0.000734 | 0.1 | 0.628 | 95.968 | 304524 | 4 | 0 |
| 0.762 | 0.703 | 10 | -7.951 | 0 | 0.306 | 0.206 | 0 | 0.0912 | 0.519 | 151.329 | 247178 | 4 | 1 |
| 0.261 | 0.0149 | 1 | -27.528 | 1 | 0.0419 | 0.992 | 0.897 | 0.102 | 0.0382 | 75.296 | 286987 | 4 | 0 |
| 0.722 | 0.736 | 3 | -6.994 | 0 | 0.0585 | 0.431 | 1.18E-06 | 0.123 | 0.582 | 89.86 | 208920 | 4 | 1 |
| 0.787 | 0.572 | 1 | -7.516 | 1 | 0.222 | 0.145 | 0 | 0.0753 | 0.647 | 155.117 | 179413 | 4 | 1 |
| 0.778 | 0.632 | 8 | -6.415 | 1 | 0.125 | 0.0404 | 0 | 0.0912 | 0.827 | 140.951 | 224029 | 4 | 1 |
| 0.666 | 0.589 | 0 | -8.405 | 0 | 0.324 | 0.555 | 0 | 0.114 | 0.776 | 74.974 | 146053 | 4 | 1 |
| 0.922 | 0.712 | 7 | -6.024 | 1 | 0.171 | 0.0779 | 3.96E-05 | 0.175 | 0.904 | 104.964 | 161800 | 4 | 1 |
| 0.794 | 0.659 | 7 | -7.063 | 0 | 0.0498 | 0.143 | 0.00224 | 0.0944 | 0.308 | 112.019 | 247460 | 4 | 0 |
| 0.853 | 0.668 | 3 | -6.995 | 1 | 0.447 | 0.263 | 0 | 0.104 | 0.745 | 157.995 | 165363 | 4 | 1 |
| 0.297 | 0.993 | 9 | -7.173 | 1 | 0.118 | 5.66E-05 | 0.77 | 0.0766 | 0.178 | 127.693 | 182427 | 4 | 0 |

## 3.4 System Design

### 3.4.1 Design Constraints

The software and hardware capabilities for this project will be determined by the capabilities of machine learning libraries and tools employed. The project will be executed on a computer with suitable hardware specifications, and any environmental setup requirements will be described.

Software requirements

| | |
|---|---|
| Operating System | Windows XP/7/8/8.1/10/11 or Linux |
| User Interface | Anaconda Navigator , Jupyter |
| Client-side Scripting | Python |
| Programming Language | Python |
| Coding Platform | VS Code |
| Connection Link | XAMPP Control Panel v3.3.0 |
| Web Browser | Google Chrome , Microsoft Edge |
| Python Libraries | NumPy , Pandas , Seaborn , Scikit-Learn , Mlatplotlib |

6

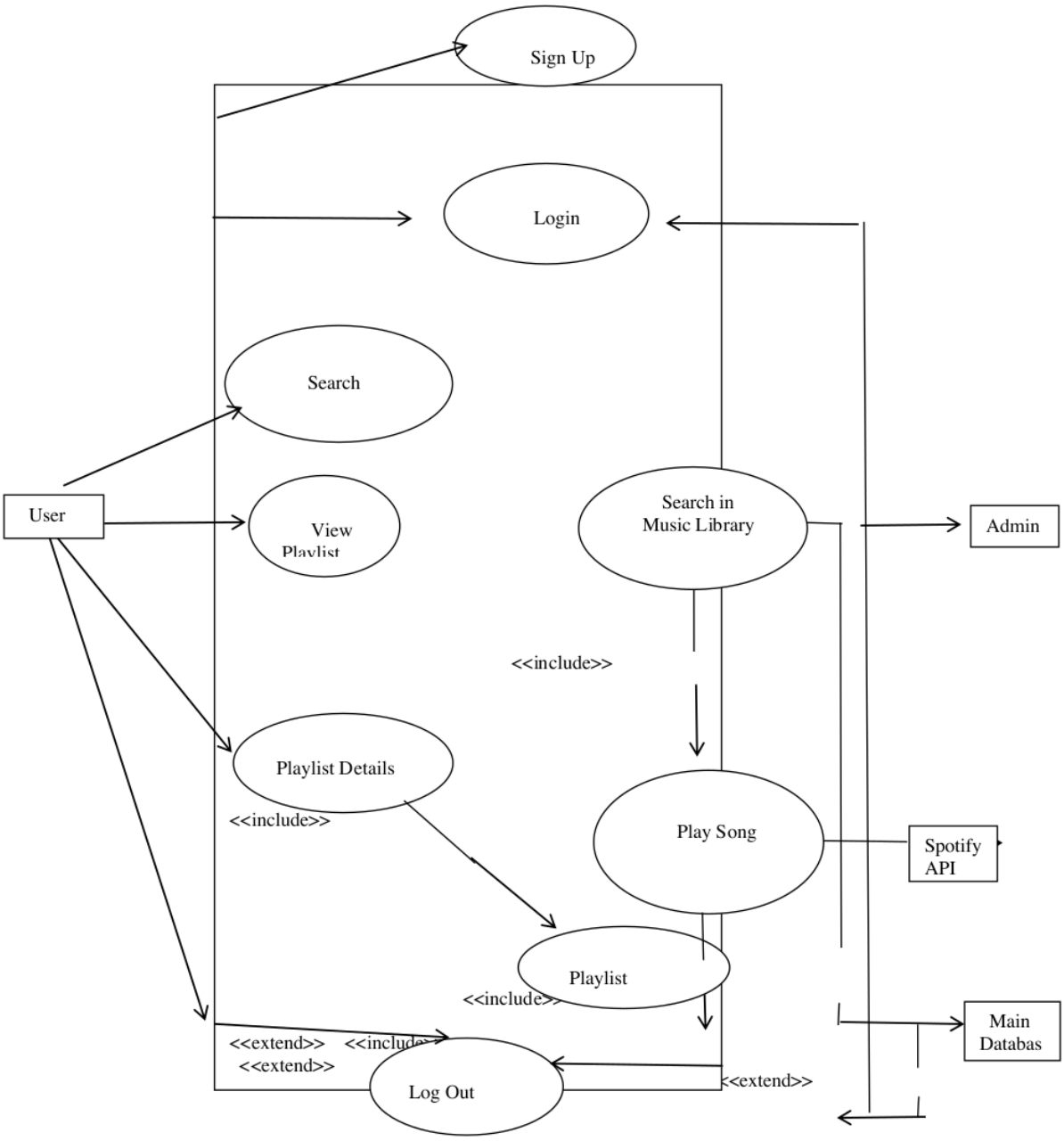
Hardware requirements

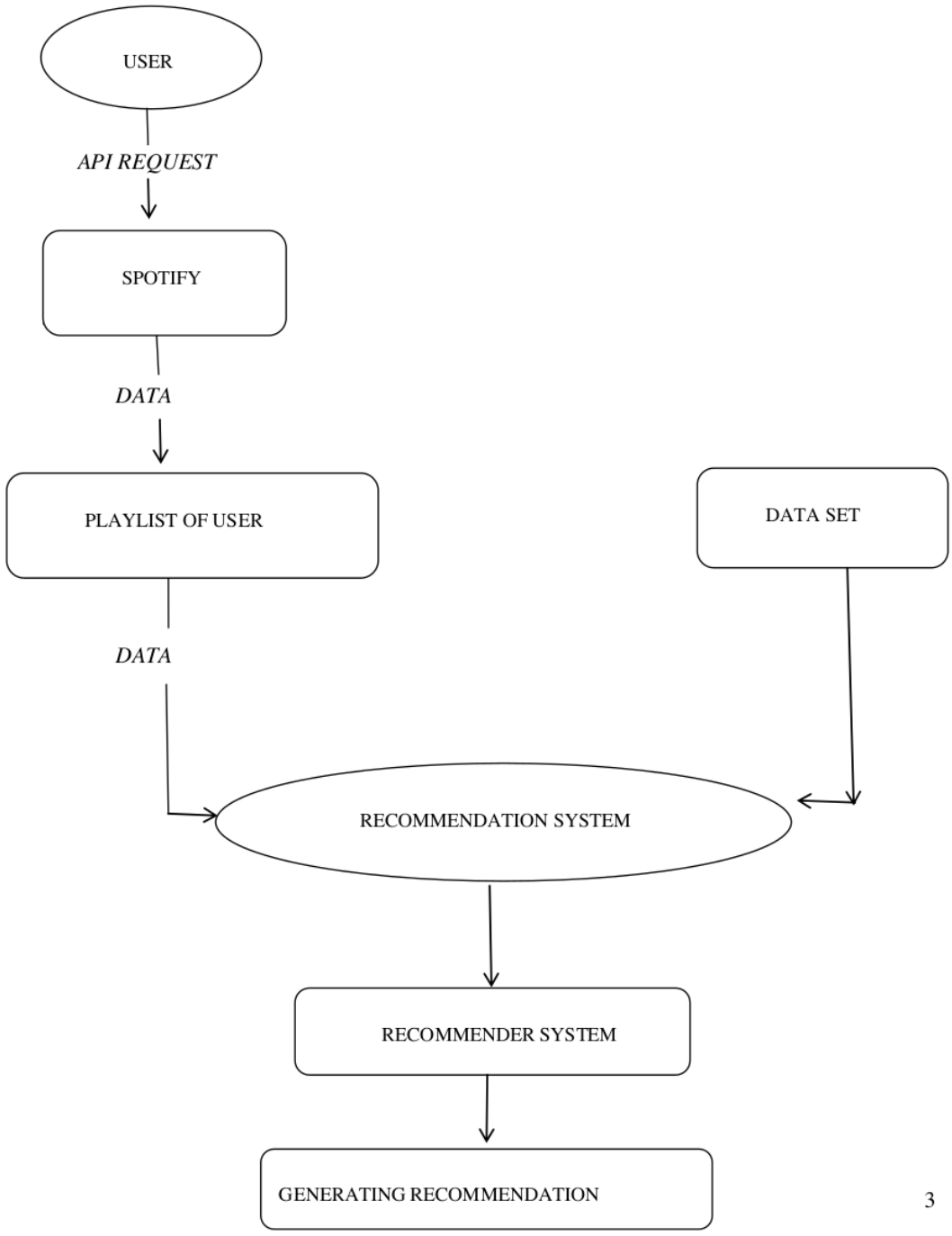
| | |
|---|---|
| Memory | 6 GB |
| Processor Type | Intel Pentium, i3, i5, i7 or faster |
| Processor Speed | 1.83 GHz or faster processor, Intel Pentium compatible |
| Swap Space | 2.3 GB |
| Hard Disk Space | 500GB/or less |
| Display | 16 Bit Color |

3.4.2 System Architecture **OR** Block Diagram

In order to propose music, the project will make use of machine learning techniques including categorization algorithms. Model training, model evaluation, and data collection and preprocessing are all part of the system design. The data flow through the systems and the many stages involved in transforming the data into predictions will be depicted in the block diagram.

MUSIC ENGINE SYSYTEM




USER
API REQUEST
SPOTIFY
DATA
PLAYLIST OF USER
DATA SET
DATA
RECOMMENDATION SYSTEM
RECOMMENDER SYSTEM
GENERATING RECOMMENDATION

# Chapter 4

# Implementation

## 4.1 Methodology OR Proposal

**Model Selection -** To train the model, many machine learning techniques are employed. The algorithms used in this project are:

1. **KNN -** A popular machine learning approach for classification and regression problems is K-Nearest Neighbors (KNN). A non-parametric, supervised learning classifier, the k-nearest neighbors (KNN) algorithm employs proximity to classify or forecast how to group a single data point.

2. **Logistic Regression -** One supervised machine learning technique that is used is termed logistic regression. It is a machine learning method for classification issues where the goal is to predict the likelihood that an instance belongs to a given class or not. The relationship between two data pieces is investigated using a statistical technique known as logistic regression.

3. **Random Forest -** A machine learning system called Random Forest combines the results of many decision trees to get a single result. Its popularity has been fueled by its adaptability, simplicity of usage, and capacity to handle both regression and classification problems.

4. **Decision Tree -** A decision tree is a kind of tree structure that is used in machine learning algorithms. It is similar to a flowchart, with leaf nodes signifying the algorithm's result, branches indicating rules, and core nodes representing features. It is a versatile supervised machine-learning technique that may be used for problems with classification and regression.

**Model Training -**

1. Logistic Regression: The logistic regression model is trained with a test size of 0.2 and random state of 42.
2. KNN: The KNN model is trained with and random state of 42.
3. Random Forest: The Random Forest model is trained with random state of 39.
4. Decision Tree**:** The Decision Tree model is trained with test size of 0.2 and random state of 42.

**Model Tuning**: The hyper parameters of the machine learning models are tuned to improve their performance. The tuned models are:

- KNN: The best hyper parameters are test size of 0.2
- Random Forest:The best hyper parameters are n_estimators=100
- Logistic Regression: The best hyper parameters are test size of 0.2
- Decision Tree: The best hyper parameters are test size of 0.21

**Model Evaluation**: The performance of the models is evaluated using the testing dataset. The evaluation metrics used are accuracy, precision, recall, F1 score and AUC.

### 4.2 Testing OR Verification Plan

Testing plays a crucial role in ensuring that the system effectively delivers personalized music recommendation to users. By rigorously testing and evaluating the system, we can ensure that it delivers accurate and relevant results.
The project is thoroughly tested and verified on various criterion of machine learning. It stands still on all criteria with more than acceptable percentages on every model. This model is tested for:

**Accuracy** - Accuracy is a metric for evaluating classification models. The total correctness of a machine learning model for classification is indicated by its accuracy. Accuracy is a metric better defined in balanced classes and care about the overall model correctness. Accuracy is the fraction of predictions our model got right or

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

Accuracy comes out to be .91 or 91% that means our model is working perfectly.

**Recall -** Recall demonstrates if a machine learning model is able to locate every object in the intended class.Accurately identifying genuine positives from each of the dataset's real positive samples serves as a proxy for recall. The number of true positives divided by the total number of positive occurrences yields the recall.

$$\text{Recall} = \frac{TP}{TP + FN}$$

The more the recall the better the model.

**Precision -** The precision of an ML model indicates the accuracy of its predictions for the target class. The number of valid positive predictions divided by the total number of positive predictions made by the model yields the precision.

$$\text{Precision} = \frac{TP}{TP + FP}$$

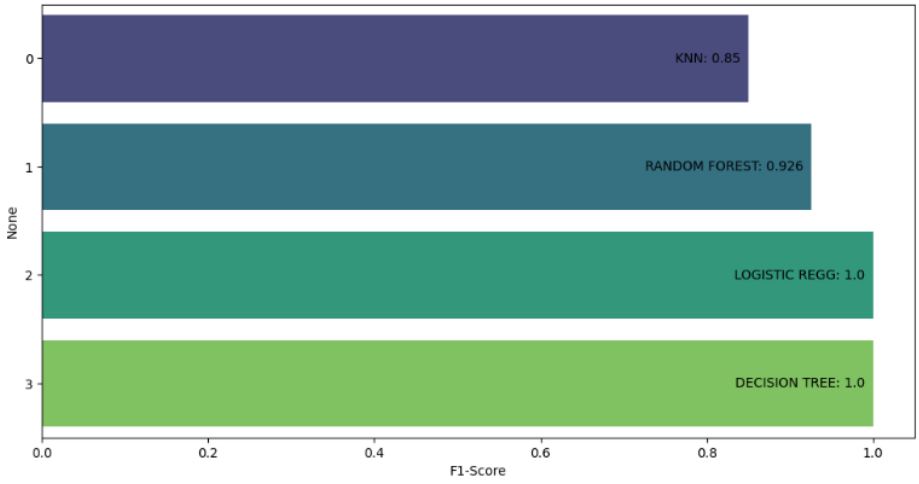
Here TP stands for true positive and FP for False Positives. The higher the precision the better the model.

**F1 score -** Instead of looking at overall performance like accuracy does, F1 score looks at a model's performance on each class separately to see how predictive it is.The F1 score has a range of 0 to 1 (or 0 to 100%), with 1 denoting the ideal outcome and 0 representing subpar performance.

**AUC -** The acronym for "Area under Curve" is AUC. Thus, the entire two-dimensional region beneath the entire Curve is measured by the AUC.
. AUC is generally used for binary classification. The higher the AUC, the better the classifier distinguishes between negative and positive.

---

**Accuracy, Recall, Precision, AUC and F1 score of:**



## KNN model -

```
from sklearn.metrics import accuracy_score
from sklearn.metrics import recall_score
from sklearn.metrics import precision_score
from sklearn.metrics import roc_auc_score
from sklearn.metrics import f1_score
accuracy = accuracy_score(y_test, y_pred)
recall = recall_score(y_test, y_pred)
precision = precision_score(y_test, y_pred)
auc = roc_auc_score(y_test, y_pred)
f1 = f1_score(y_test, y_pred)
print("Accuracy: {:.2f}%".format(accuracy*100))
print("Recall: {:.2f}%".format(recall*100))
print("Precision: {:.2f}%".format(precision*100))
print("F1 Score: {:.3f}%".format(f1*100))
print("AUC: {:.2f}%".format(auc*100))
```

```
Accuracy: 84.62%
Recall: 85.00%
Precision: 85.00%
F1 Score: 85.000%
AUC: 84.61%
```

## Random Forest -

```
from sklearn.metrics import accuracy_score
from sklearn.metrics import recall_score
from sklearn.metrics import precision_score
from sklearn.metrics import roc_auc_score
from sklearn.metrics import f1_score
accuracy = accuracy_score(y_test, predictions_forest )
recall = recall_score(y_test, predictions_forest )
precision = precision_score(y_test, predictions_forest )
auc = roc_auc_score(y_test, predictions_forest )
f1 = f1_score(y_test, predictions_forest )
print("Accuracy: {:.2f}%".format(accuracy*100))
print("Recall: {:.2f}%".format(recall*100))
print("Precision: {:.2f}%".format(precision*100))
print("F1 Score: {:.2f}%".format(f1*100))
print("AUC: {:.2f}%".format(auc*100))
```

```
Accuracy: 92.31%
Recall: 95.00%
Precision: 90.48%
F1 Score: 92.68%
AUC: 92.24%
```

## Logistic Regression -

```
from sklearn.metrics import accuracy_score, recall_score, precision_score, roc_auc_score, f1_score

accuracy = accuracy_score(y_test, predictions_logistic)
recall = recall_score(y_test, predictions_logistic)
precision = precision_score(y_test, predictions_logistic)
auc = roc_auc_score(y_test, predictions_logistic)
f1 = f1_score(y_test, predictions_logistic)

print("Accuracy: {:.2f}%".format(accuracy * 100))
print("Recall: {:.2f}%".format(recall * 100))
print("Precision: {:.2f}%".format(precision * 100))
print("F1 Score: {:.2f}%".format(f1 * 100))
print("AUC: {:.2f}%".format(auc * 100))
```

```
Accuracy: 100.00%
Recall: 100.00%
Precision: 100.00%
F1 Score: 100.00%
AUC: 100.00%
```

## Decision Tree -

```
accuracy = accuracy_score(y_test, predictions)
recall = recall_score(y_test, predictions)
precision = precision_score(y_test, predictions)
auc = roc_auc_score(y_test, predictions)
f1 = f1_score(y_test, predictions)
print("Accuracy: {:.2f}%".format(accuracy * 100))
print("Recall: {:.2f}%".format(recall * 100))
print("Precision: {:.2f}%".format(precision * 100))
print("F1 Score: {:.2f}%".format(f1 * 100))
print("AUC: {:.2f}%".format(auc * 100))
```

```
Accuracy: 100.00%
Recall: 100.00%
Precision: 100.00%
F1 Score: 100.00%
AUC: 100.00%
```

**ROC -** The receiver operating characteristic curve, or ROC curve, is a graph that displays a classification model's performance over all categorization levels.Two parameters are plotted on the curve:

True Positive Rate

$$TPR = \frac{TP}{TP + FN}$$

False Positive Rate

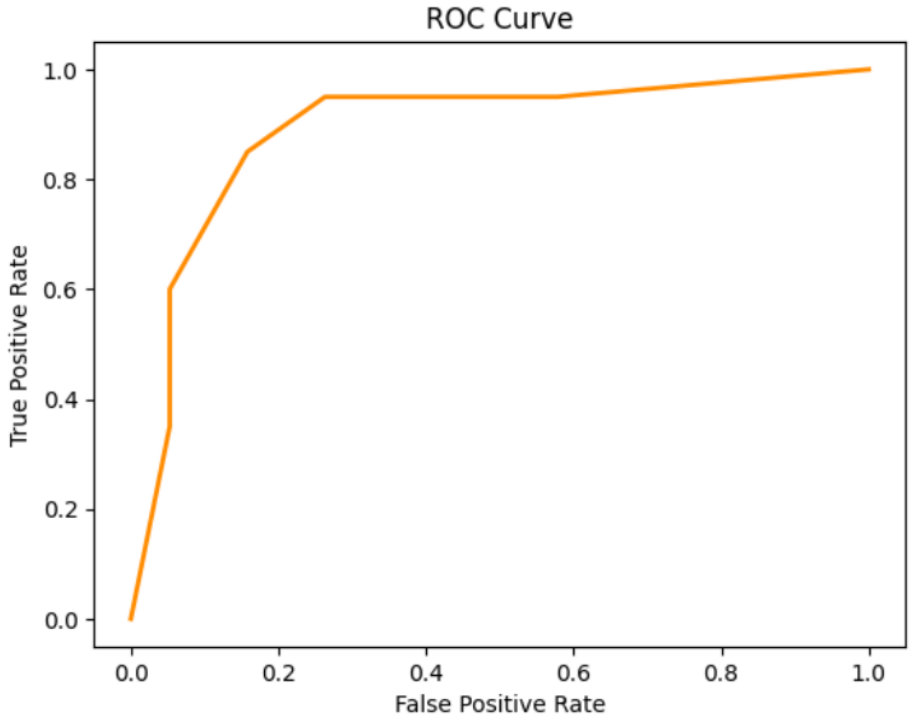
$$FPR = \frac{FP}{FP + TN}$$

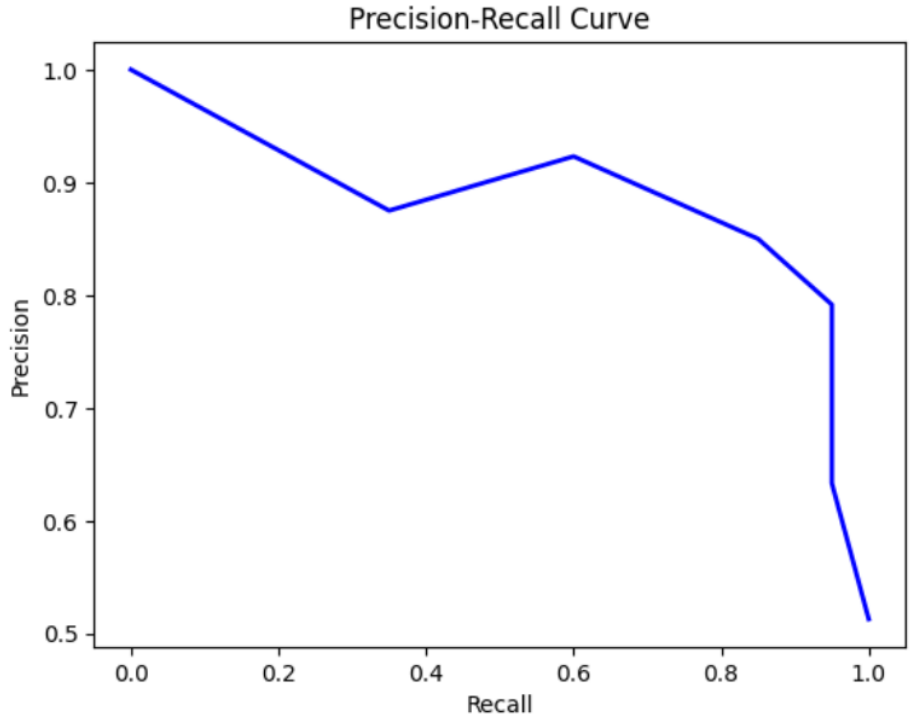
**Precision-Recall Curve -** The precision-recall curve shows the recall vs. accuracy trade-off for different thresholds. Low false positive and false negative rates are associated with good accuracy and high recall, respectively. High accuracy and recall are both indicated by an area under the curve with a high value.

---

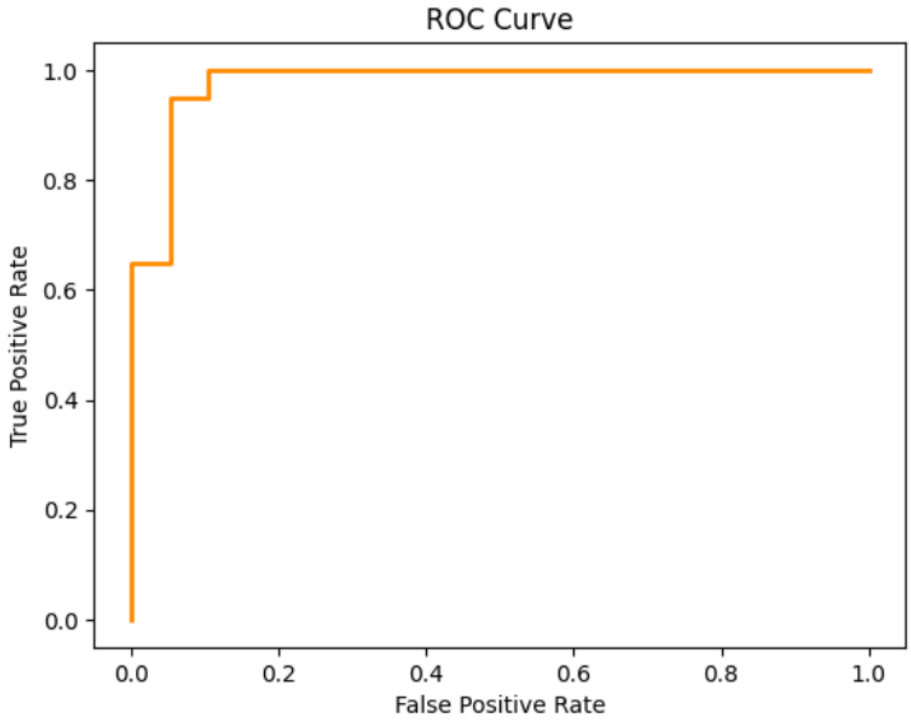
**ROC and Precision-Recall Curve of:**

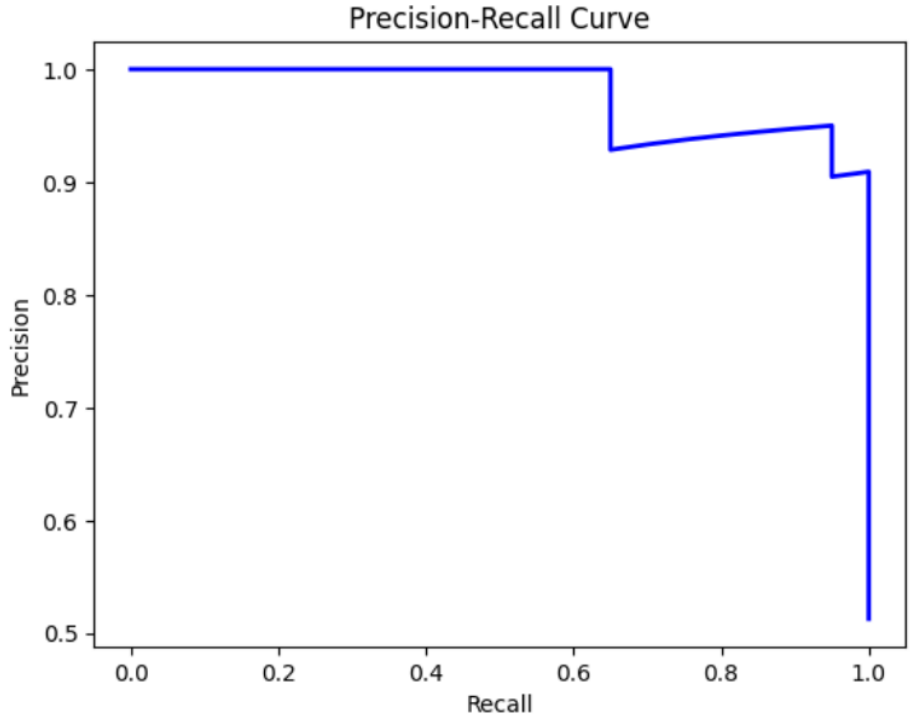
**KNN -**


ROC Curve
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0

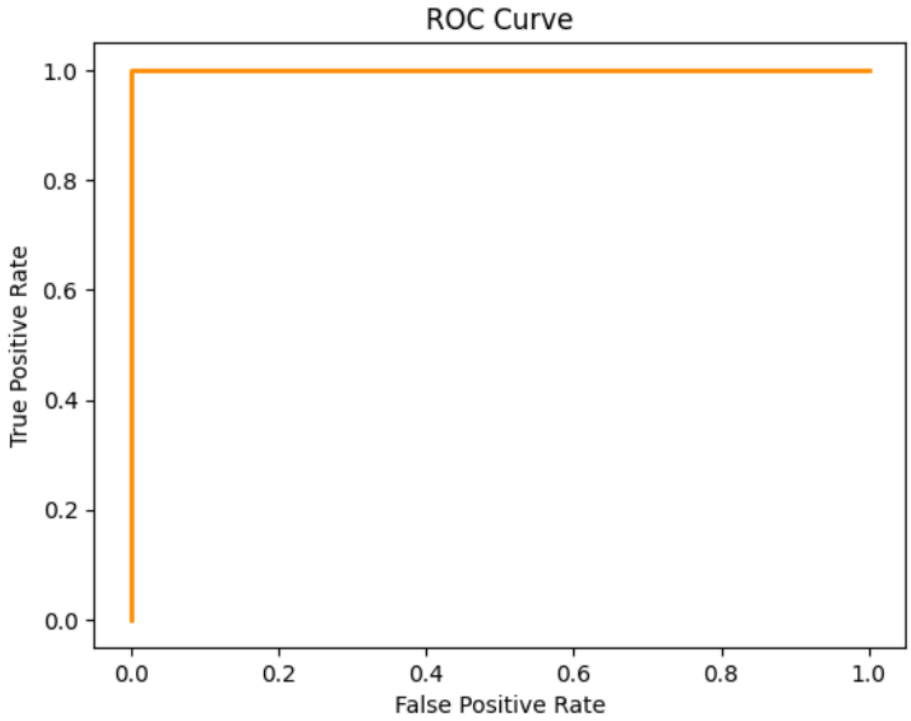

Precision-Recall Curve
Precision
Recall
0.5
0.6
0.7
0.8
0.9
1.0
0.0
0.2
0.4
0.6
0.8
1.0

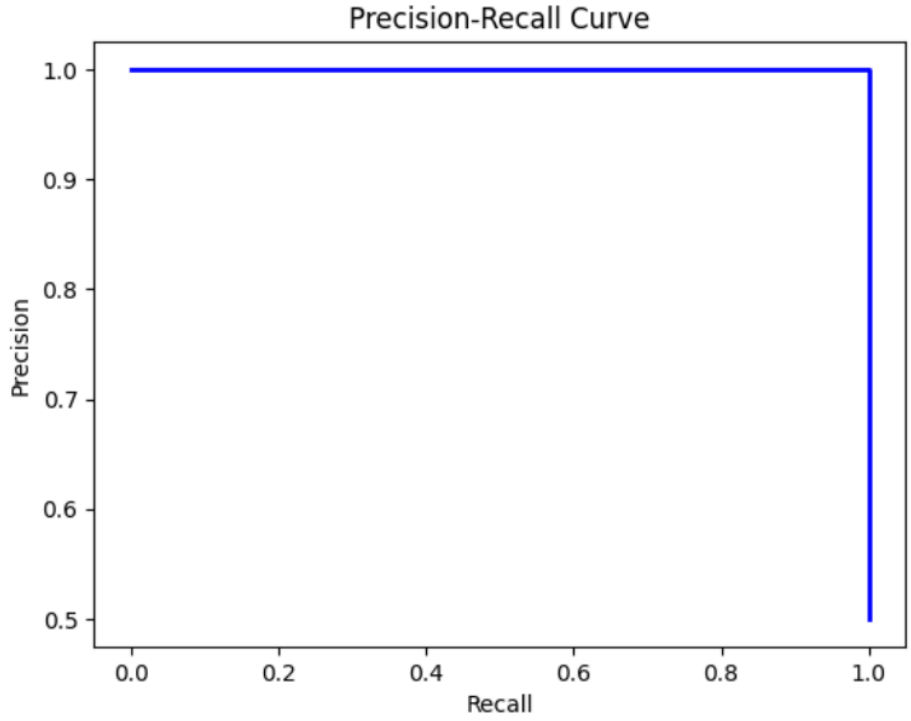
**Random Forest -**



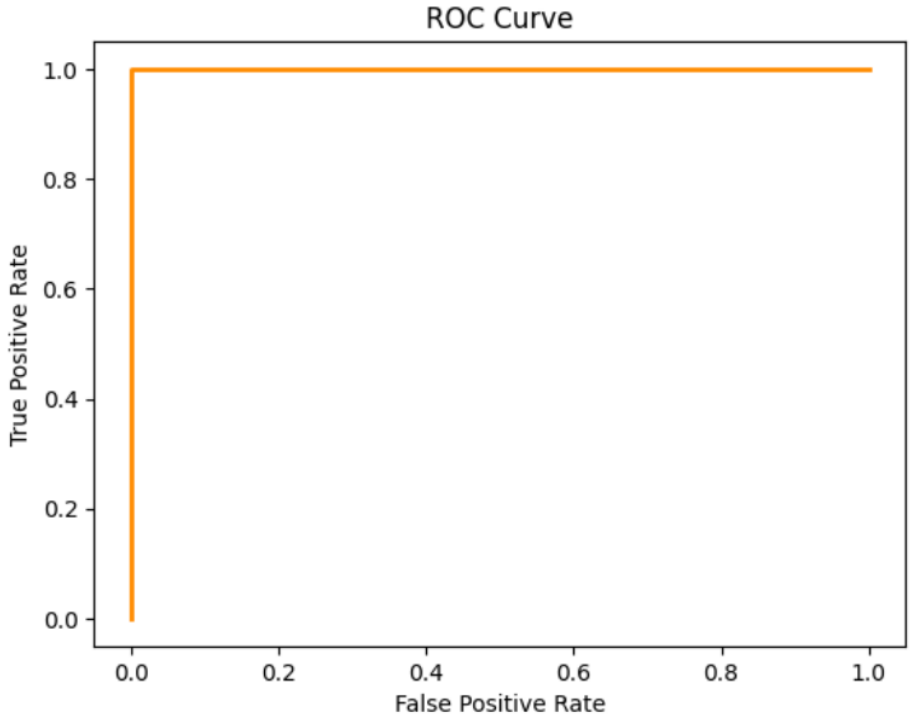



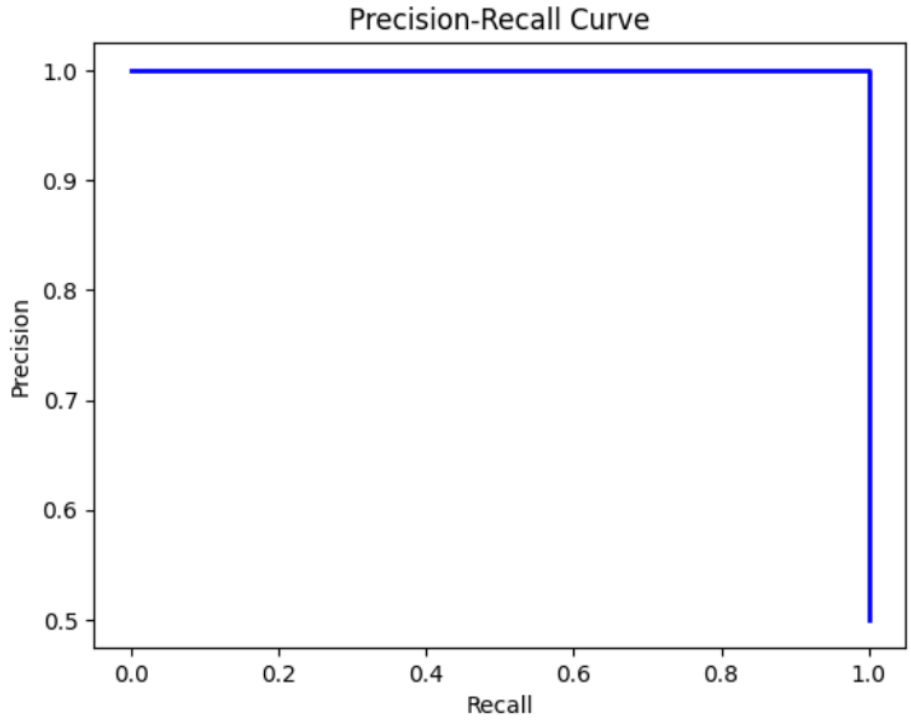
**Logistic Regression -**



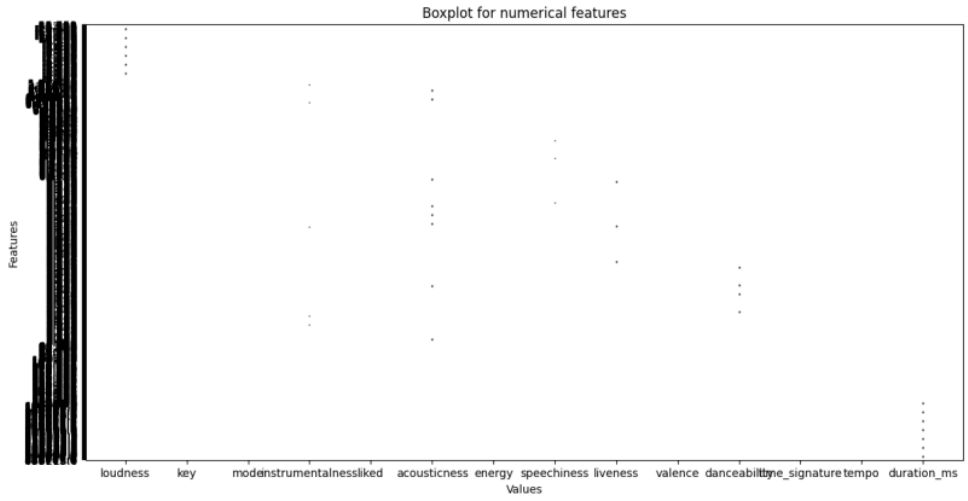



**Decision Tree -**

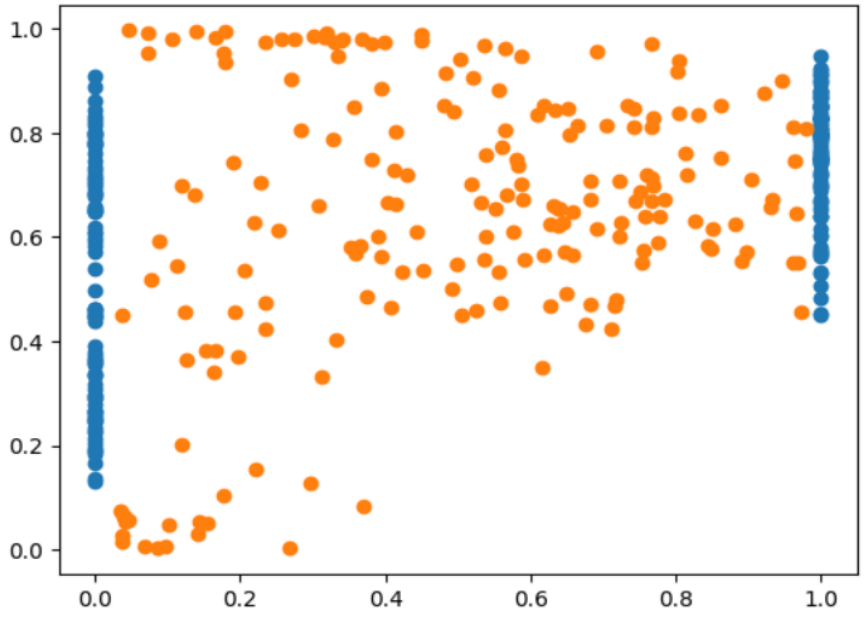


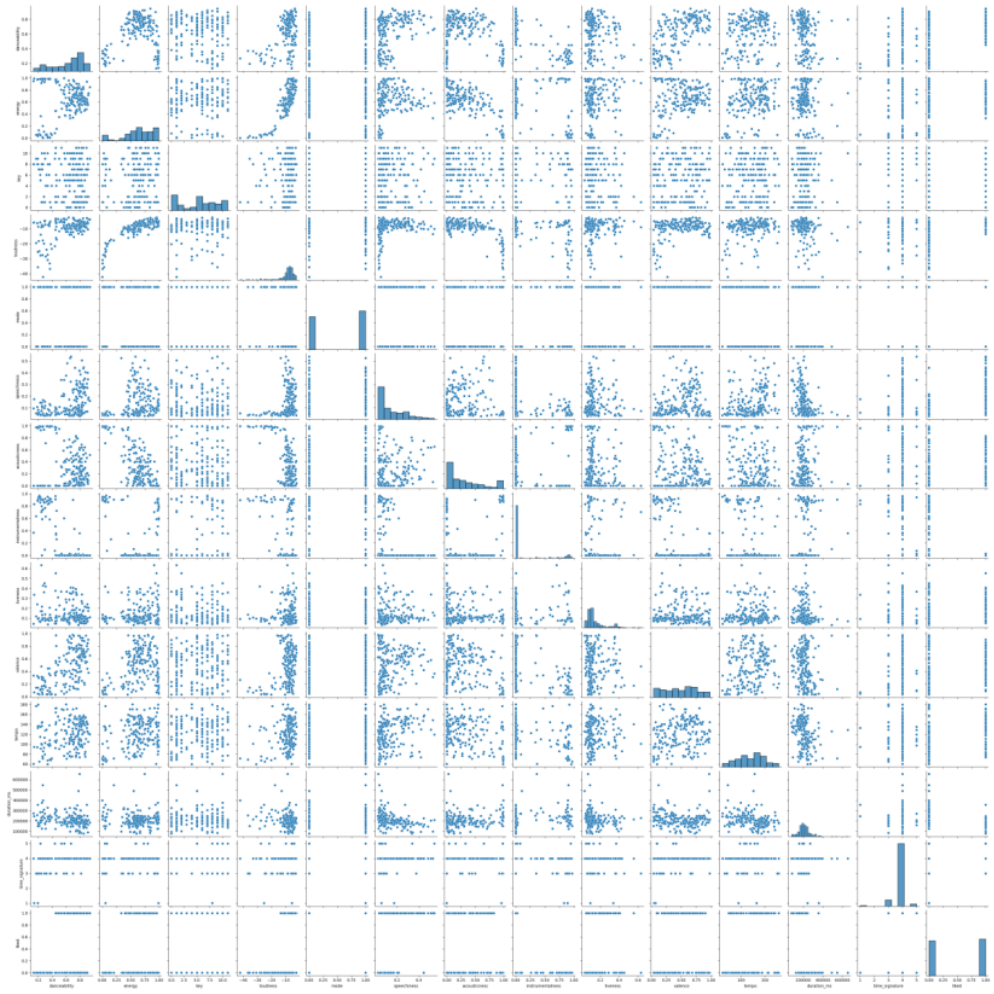
**Box Plot -** The minimum value, first quartile, median, third quartile, and maximum value are all displayed graphically in a box plot of a data set.



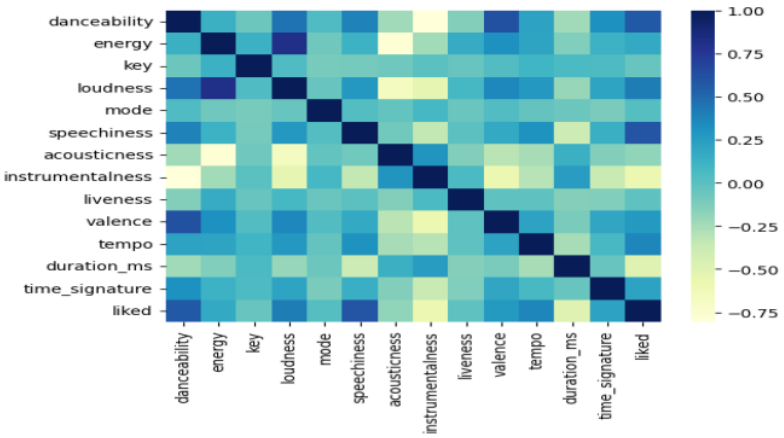

**Scatter Plot -** A scatter plot, also known as a scatter graph, is a kind of diagram that shows the values of two variables for a collection of data using Cartesian coordinates. Scatter plots are used to observe relationships between variables.



**Pair Plot -** The Seaborn Pairplot may be used to plot pairwise relationships between variables within a dataset. This provides an excellent representation and facilitates our comprehension of the data by condensing a large amount of information into a single image.

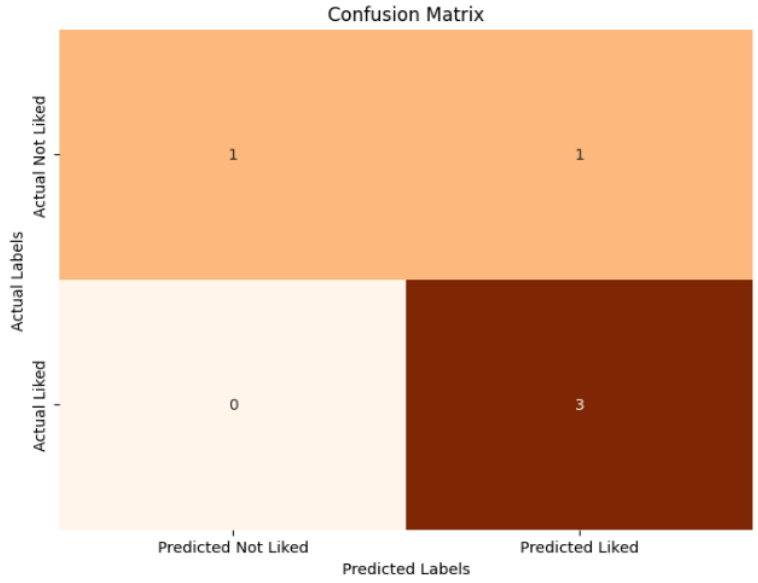


**Heatmap** -A basic heat map allows users to quickly identify the most important or relevant data points. A heat map is a two-dimensional data representation where difference across two axes is displayed.

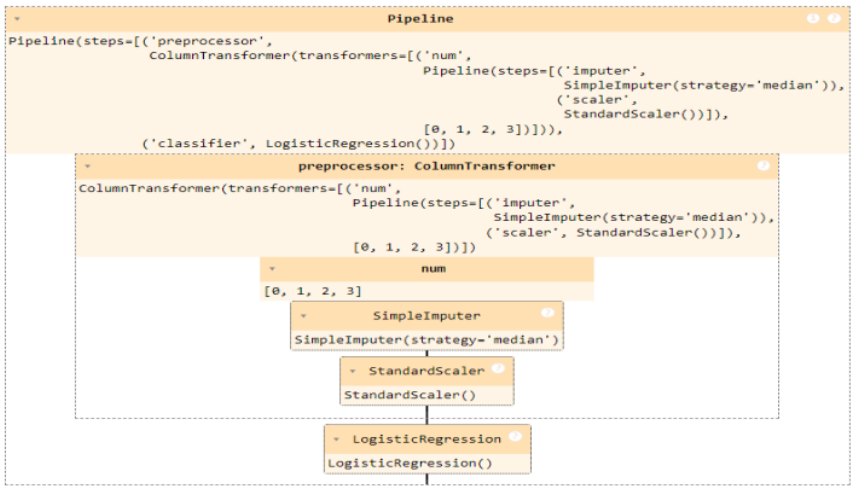



18

**Confusion Matrix -** Confusion Matrix, often referred to as an error matrix, is a particular table arrangement that makes algorithm performance visible. Confusion matrix is supervised learning.The performance of a classification algorithm is summarized and shown using a confusion matrix.



**Pipeline** - Building, training, testing, and deploying machine learning models may be done more quickly, consistently, and automatically with the use of a machine learning pipeline, which is a linked set of data processing and modeling operations.






# Chapter 5

# Standards Adopted

| Category | Coding Standards |
|---|---|
| **General Coding Standards** | Use consistent indentation style, naming conventions, comments, data types, input validation, error handling, version control |
| **Jupyter Standards** | Use Headings and Markdown Lavishly.<br>Structure of Your Data Science Notebook. |
| **Machine Learning Standards** | Develop & Document Model Training Metrics<br>Fine Tune The Serving ML Model |
| **Accessibility Coding Standards** | Use semantic Python tags, appropriate color contrast, keyboard navigation |
| **Security Coding Standards** | Use secure web server, validate user input, use secure cookies, hash and salt passwords, use WAF |
| **XAMPP Coding Standards** | Follow XAMPP's coding standards, use XAMPP's built-in tools for linting and testing |
| **Additional Coding Standards** | Follow project's coding standards, use code review process, continuously improve coding skills |

# Chapter 6

# Conclusion and Future Scope

## 6.1 Conclusion

Our conclusions, derived from the experiment, are as follows. To improve the caliber of music recommendations, the music recommender system should first take the genre information into account. Second, the overall performance of CRNNs that take into account both frequency characteristics and time sequence patterns is superior. It demonstrates how well its hybrid construction extracts the musical elements. Based on our analysis, we may recommend that additional music elements be added in future study to increase the recommender system's accuracy. For example, tempo gram might be used to capture the local tempo at a certain period.

## 6.2 Future Scope

With data analysis and technological breakthroughs driving the field, music reco mmendation systems have a bright future ahead of them.
The following are some possible avenues for improvement and growth:

Personalisation: As musical tastes and situations change, music suggestion algori thms will become increasingly more customised.
Recommendation algorithms can provide highly personalised suggestions if they incorporate additional user data, such as listening history, mood, location, and a ctivity.

Sensitive to context Suggestions:
In the future, computers will use contextual data to offer suggestions that are ap propriate for the user's present circumstances.
The time of day, the weather, the social context, and even physiological informat ion like stress levels or heart rate might be included in
this.

## ***References***

Aiolli, F. (2012). A preliminary study on a recommender system for the million songs dataset challenge. PREFERENCE LEARNING: PROBLEMS AND APPLICATIONS IN AI

2 Spotify. (2013, March) Spotify Music for everyone. [Online].

3 Peter Knees and Markus Schedl. Music retrieval and recommendation – a tutorial overview. In Proceedings of the 38th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), Santiago, Chile, August 2015.

4 Peter Knees and Markus Schedl. Music Similarity and Retrieval: An Introduction to Audio- and Web-based Strategies. 2016.

5 Chris Johnson. From idea to execution: Spotify's discover weekly. November 2015.

[6] Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions, Gediminas Adomavicius, Alexander Tuzhilin, IEEE Members

[7] Creating a Hybrid Music Recommendation System from Content and Social-Based Algorithms Jamie Cai, John Francis, Stephen Gheysens, Rutgers University, GSET „09

[8] http://en.wikipedia.org/wiki/Acoustic_fingerprint, Sun, 24-10-2010

[9] http://marsyas.info, Sun 24-10-2010

## INDIVIDUAL CONTRIBUTION REPORT:

# MUSIC RECOMMENDATION SYSTEM

ALKESH SINGH
2129021

**Abstract:** The aim of a spotify music recommendation system is to provide convenient, efficient, and diverse music experiences. Objectives include ensuring a user-friendly interface, a wide song selection, and an effective recommendation system. User engagement and mobile responsiveness are crucial for customer retention. Marketing efforts aim to build brand awareness, while data analytic inform business decisions. Sustainable practices, adaptability, and innovation contribute to long-term success in the dynamic market. Overall, the focus is on meeting customer needs, and staying competitive through continuous improvement and strategic initiatives.

**Individual contribution and findings:** Throughout the entire project, I helped my team in gathering data in CSV format (i.e., data.csv) and data mining processess to discover patterns,trends and insights from large datasets.I've also made the code and passed the test cases for Decision tree method found it's accuracy percentages and gave several ideas and codes which could help my team to run the code.I've used python language for coding and in the end made the pipeline model of the recommendation system.Pipeline model in the ending part is also written by me.

**Individual contribution to project report preparation:** For the report preperation I've done the future scope,references that we've taken from during our project formation and data gathering and forming the decision tree part in the report.

**Individual contribution for project presentation and demonstration:** I've helped my team in making the slides of future scope,references and conclusion part is done by me during presentation (ppt.x) formation.

Full Signature of Supervisor:

Full Signature of the student: Alkesh Singh

……..…

……..

## SAMPLE INDIVIDUAL CONTRIBUTION REPORT:

## MUSIC RECOMMENDATION SYSTEM

Pradyumn Mukherjee
2129082

**Abstract:** The aim of a spotify music recommendation system is to provide convenient, efficient, and diverse music experiences. Objectives include ensuring a user-friendly interface, a wide song selection, and an effective recommendation system. User engagement and mobile responsiveness are crucial for customer retention. Marketing efforts aim to build brand awareness, while data analytic inform business decisions. Sustainable practices, adaptability, and innovation contribute to long-term success in the dynamic market. Overall, the focus is on meeting customer needs, and staying competitive through continuous improvement and strategic initiatives.

**Individual contribution and findings:** Throughout the entire project, I helped my team in data normalization or standardization for ensuring that each feature should contribute equally in the model by scaling it to a comparable range.I've also made the code and passed the test cases for KNN(K- NEAREST NEIGHBOUR) method found it's accuracy percentages and gave several ideas and codes which could help my team to run the code.I've used python language for coding and in the end made the box plot of the recommendation system using jupyter framework.

**Individual contribution to project report preparation:** For the report preperation I've done the System design part which includes design constraints and System Architecture with UML block diagram during our project formation and data gathering part in the report.

**Individual contribution for project presentation and demonstration:** I've helped my team in making the slides of analysis of various models and challenges part is done by me during presentation (ppt.x) formation.

Full Signature of Supervisor:

Full Signature of the student: Pradyumn Mukherjee

……..…

……..

## INDIVIDUAL CONTRIBUTION REPORT:

## MUSIC RECOMMENDATION SYSTEM

Mukul Jain
2129078

**Abstract:** The aim of a spotify music recommendation system is to provide convenient, efficient, and diverse music experiences. Objectives include ensuring a user-friendly interface, a wide song selection, and an effective recommendation system. User engagement and mobile responsiveness are crucial for customer retention. Marketing efforts aim to build brand awareness, while data analytic inform business decisions. Sustainable practices, adaptability, and innovation contribute to long-term success in the dynamic market. Overall, the focus is on meeting customer needs, and staying competitive through continuous improvement and strategic initiatives.

**Individual contribution and findings:** Throughout the entire project, I helped my team in data cleaning by finding addressing outliers,inconsistent data, and missing values in the dateset to guarantee its accuracy and dependency.I've also made the code and passed the test cases for Logistic Regression model found it's accuracy percentages and gave several ideas and codes which could help my team to run the code.I've used python language for coding and in the end made the Scatter Plot and Heatmap of the recommendation system.

**Individual contribution to project report preparation:** For the report preparation I've done the Standard adoption and implementation of with methodology,testing,result analysis and quality assurance part from the project in the report.

**Individual contribution for project presentation and demonstration:** I've helped my team in making the slides for setting up agenda,proposed timeline and framework part is done by me during presentation (ppt.x) formation.

Full Signature of Supervisor:

Full Signature of the student: Mukul Jain

……..….

……..

## INDIVIDUAL CONTRIBUTION REPORT:

### MUSIC RECOMMENDATION SYSTEM

Ashutosh Dubey
2129128

**Abstract:** The aim of a spotify music recommendation system is to provide convenient, efficient, and diverse music experiences. Objectives include ensuring a user-friendly interface, a wide song selection, and an effective recommendation system. User engagement and mobile responsiveness are crucial for customer retention. Marketing efforts aim to build brand awareness, while data analytic inform business decisions. Sustainable practices, adaptability, and innovation contribute to long-term success in the dynamic market. Overall, the focus is on meeting customer needs, and staying competitive through continuous improvement and strategic initiatives.

**Individual contribution and findings:** Throughout the entire project, I helped my team in data preparation by turning unprocessed data into processing one and done dataa encoding by converting category value into numerical representation .I've also made the code and passed the test cases for Random Forest model found it's accuracy percentages and gave several ideas and codes which could help my team to run the code. I've used python language for coding and in the end made the confuion matrix and pair plot of the recommendation system.

**Individual contribution to project report preparation:** For the report preperation I've done the introduction,literature review and problem statement that we've taken from during our project formation and data gathering and forming the decision tree part in the report.

**Individual contribution for project presentation and demonstration:** I've helped my team in making the slides of problem statement,methodologies and several types of methods with brief details and conclusion part is done by me during presentation (ppt.x) formation.

Full Signature of Supervisor:

Full Signature of the student: Ashutosh Dubey

……..…

……..

# MUSIC RECOMMENDATION SYSTEM

ORIGINALITY REPORT

| 23% | 18% | 4% | 15% |
|---|---|---|---|
| SIMILARITY INDEX | INTERNET SOURCES | PUBLICATIONS | STUDENT PAPERS |

PRIMARY SOURCES

| | | |
|---|---|---|
| 1 | www.coursehero.com<br>Internet Source | 4% |
| 2 | www.worldleadershipacademy.live<br>Internet Source | 4% |
| 3 | www.ir.juit.ac.in:8080<br>Internet Source | 3% |
| 4 | Submitted to CSU, Fullerton<br>Student Paper | 1% |
| 5 | www.politesi.polimi.it<br>Internet Source | 1% |
| 6 | www.slideshare.net<br>Internet Source | 1% |
| 7 | Submitted to Banaras Hindu University<br>Student Paper | 1% |
| 8 | Submitted to University of Wales Institute, Cardiff<br>Student Paper | 1% |
| 9 | Submitted to KIIT University<br>Student Paper | 1% |

| | | |
|---|---|---|
| 10 | iq.opengenus.org<br>Internet Source | 1 % |
| 11 | Submitted to National College of Ireland<br>Student Paper | <1 % |
| 12 | Submitted to Myanmar Imperial College<br>Student Paper | <1 % |
| 13 | Submitted to Swiss School of Business and Management - SSBM<br>Student Paper | <1 % |
| 14 | Submitted to University of Wisconsin Extension<br>Student Paper | <1 % |
| 15 | Submitted to University of Surrey<br>Student Paper | <1 % |
| 16 | Submitted to Wilmington University<br>Student Paper | <1 % |
| 17 | Submitted to Campbellsville University<br>Student Paper | <1 % |
| 18 | Submitted to University of East London<br>Student Paper | <1 % |
| 19 | Submitted to Westcliff University<br>Student Paper | <1 % |
| 20 | Submitted to INTI Universal Holdings SDM BHD<br>Student Paper | <1 % |

| | | |
|---|---|---|
| 21 | Submitted to University of Colombo<br>Student Paper | <1 % |
| 22 | dspace.univ-ouargla.dz<br>Internet Source | <1 % |
| 23 | eurchembull.com<br>Internet Source | <1 % |
| 24 | repository.uantwerpen.be<br>Internet Source | <1 % |
| 25 | pdfs.semanticscholar.org<br>Internet Source | <1 % |
| 26 | github.com<br>Internet Source | <1 % |
| 27 | oer.calebuniversity.edu.ng<br>Internet Source | <1 % |
| 28 | Wang Xia, Ge Hongwei, Zhao Mingde, Hou Yaqing, Sun Mingyang. "Stochastic online decisioning hyper-heuristic for high dimensional optimization", Applied Intelligence, 2023<br>Publication | <1 % |

| | | | |
|---|---|---|---|
| Exclude quotes | Off | Exclude matches | Off |
| Exclude bibliography | Off | | |