Dissertation on

“Music Recommender System With Sentiment Analysis”

Submitted in partial fulfillment of the requirements for the award of degree of

Bachelor of Technology
in
Computer Science & Engineering

UE20CS461A – Capstone Project Phase - 2

Submitted by:

Reshmi Pradeep	**PES2UG20CS270**
Alekhya Sundari R Nanduri	**PES2UG20CS900**
D Mrudula	**PES2UG20CS905**
Harshitha Golla	**PES2UG20CS910**

Under the guidance of

Dr. L Kamatchi Priya
Associate Professor
PES University

June - Nov 2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
FACULTY OF ENGINEERING
PES UNIVERSITY
(Established under Karnataka Act No. 16 of 2013)
Electronic City, Hosur Road, Bengaluru – 560 100, Karnataka, India

PES UNIVERSITY

(Established under Karnataka Act No. 16 of 2013)
Electronic City, Hosur Road, Bengaluru – 560 100, Karnataka, India

FACULTY OF ENGINEERING

CERTIFICATE

This is to certify that the dissertation entitled

'Music Recommender System With Sentiment Analysis'

is a bonafide work carried out by

Reshmi Pradeep	**PES2UG20CS270**
Alekhya Sundari R Nanduri	**PES2UG20CS900**
D. Mrudula	**PES2UG20CS905**
Harshitha Golla	**PES2UG20CS910**

In partial fulfillment for the completion of seventh semester Capstone Project Phase - 2 (UE20CS461A) in the Program of Study - Bachelor of Technology in Computer Science and Engineering under rules and regulations of PES University, Bengaluru during the period June 2023 – Nov. 2023. It is certified that all corrections / suggestions indicated for internal assessment have been incorporated in the report. The dissertation has been approved as it satisfies the 7th semester academic requirements in respect of project work.

Signature	Signature	Signature
Dr. L Kamatchi Priya	Dr. Sandesh B J	Dr. B K Keshavan
Associate Professor	Chairperson	Dean of Faculty

External Viva

Name of the Examiners **Signature with Date**

1. ____________________________ ____________________________

2. ____________________________ ____________________________

DECLARATION

We hereby declare that the Capstone Project Phase - 2 entitled **“Music Recommender System with Sentiment Analysis”** has been carried out by us under the guidance of Dr. L. Kamatchi Priya, Associate Professor and submitted in partial fulfilment of the course requirements for the award of degree of **Bachelor of Technology** in **Computer Science and Engineering** of **PES University, Bengaluru** during the academic semester June – Nov. 2023. The matter embodied in this report has not been submitted to any other university or institution for the award of any degree.

PES2UG20CS270	**Reshmi Pradeep**	
PES2UG20CS900	**Alekhya Sundari R Nanduri**	Alekhya
PES2UG20CS905	**D. Mrudula**	
PES2UG20CS910	**Harshitha Golla**	Harshitha

ACKNOWLEDGEMENT

I would like to express my gratitude to Dr. L Kamatchi Priya, Department of Computer Science and Engineering, PES University, for her continuous guidance, assistance, and encouragement throughout the development of this UE20CS461A - Capstone Project Phase – 2.

I am grateful to the Capstone Project Coordinator, Dr. Sarasvathi V, Professor and Dr. Sudeepa Roy Dey, Associate Professor, for organizing, managing, and helping with the entire process.

I take this opportunity to thank Dr. Sandesh B J, Chairperson, Department of Computer Science and Engineering, PES University, for all the knowledge and support I have received from the department. I would like to thank Dr. B.K. Keshavan, Dean of Faculty, PES University for his help.

I am deeply grateful to Dr. M. R. Doreswamy, Chancellor, PES University, Prof. Jawahar Doreswamy, Pro Chancellor – PES University, Dr. Suryaprasad J, Vice-Chancellor, PES University and Prof. Nagarjuna Sadineni, Pro-Vice Chancellor - PES University, for providing to me various opportunities and enlightenment every step of the way. Finally, this project could not have been completed without the continual support and encouragement I have received from my family and friends.

ABSTRACT

The purpose of this project is to create a music recommendation system that classifies the emotions conveyed in Twitter user tweets using machine learning and natural language processing techniques, and then uses a hybrid content-based and collaborative filtering approach to suggest appropriate music.

The project mines user tweets for useful features using cutting-edge methods including term frequency-inverse document frequency (TF-IDF), GloVe embeddings, and Universal Sentence Encoder . After that, a deep convolutional neural network uses these features as input to classify emotions. The DCNN learns to categorize tweets into several emotion categories, such as joy, sadness, rage, etc., using a labeled dataset of tweets annotated with emotion labels. Once the emotions have been identified, the project uses a hybrid strategy that blends collaborative filtering with content-based recommendation to produce tailored music recommendations using the emotional context that was retrieved from tweets.

The proposed strategy is tested in the project using a sizable dataset of tweets and music data, and the findings indicate promising gains in music recommendation accuracy when compared to conventional approaches. By adding the emotional context of user tweets, the suggested methodology has the potential to improve the user experience of music recommendation systems. It also makes a contribution to the fields of NLP and machine learning by investigating cutting-edge methods for music recommendation utilizing social media data.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Machine Learning and NLP

In the science of artificial intelligence (AI), two of the most crucial areas are machine learning (ML) and natural language processing (NLP). These technologies enable us to develop intelligent systems that can recognize, evaluate, and produce language that is similar to that of humans. NLP gives these systems the ability to work with textual data, including speech, natural language text, and other types of unstructured data, while ML enables these systems to learn and enhance their performance over time. There are several uses for ML and NLP, including sentiment analysis, chatbots, virtual assistants, and content recommendation systems. In order to create an emotion-based music recommendation system that can automatically propose music based on the user's emotional state, we harness the power of ML and NLP in this project. We aim to develop a system that can give users a more personalized and interesting music listening experience by utilizing the most recent developments in ML and NLP.

1.1.1 Emotion Classification

A key component of human behavior and communication is emotion. In a number of disciplines, including psychology, sociology, and artificial intelligence, it is crucial to comprehend and understand emotions. Natural language processing and machine learning researchers are actively working on the challenge of emotion classification, which involves giving a label or category to a text or spoken utterance that accurately captures the emotion represented in it. It can be used for a variety of things, including social media sentiment analysis and tailored music suggestions. The ability to categorize emotions in texts or voice can make it possible for machines to understand human emotions and react accordingly, giving users a more individualized and sympathetic experience. Researchers are continually experimenting with novel methods and strategies to enhance the performance of emotion categorization, which has become more accurate thanks to developments in deep learning and neural networks. To achieve high accuracy in predicting users' emotional states based on their social media behavior, we present a novel method for emotion classification in this project that combines the strengths of multiple NLP and ML techniques.

1.1.2 Music Recommenders

The use of music recommendation systems, which let users find new musicians and songs based on their musical preferences, has grown in popularity in recent years. These systems analyse music data using sophisticated algorithms and offer the user personalized recommendations. Collaborative filtering is the foundation of music recommenders, which base their recommendations on a user's historical listening habits as well as those of other users who have similar listening habits. On the other hand, hybrid recommenders combine the collaborative filtering strategy with content-based filtering, which entails evaluating music qualities including genre, tempo, and mood. In this project, we aim to build a hybrid music recommender system that takes into account the user's emotional state, using machine learning and natural language processing techniques to classify the emotions of the user's tweets and recommend music accordingly.

CHAPTER 2

PROBLEM STATEMENT

The project's goal is to create a music recommendation system that accurately categorizes the emotions conveyed in tweets using Machine Learning and Natural Language Processing (NLP) methods and hence provide users with a personalized playlist. Due to the widespread use of social media, tweets frequently include extensive information on the feelings, beliefs, and experiences of users. Tweets also tend to be uploaded in the spur of a moment allowing a user's current emotional state and personality to shine through. The project's objective is to use this data to provide tailored music suggestions based on the sentiments shared by individuals in their tweets. The goal of the research is to tackle the difficulty of effectively identifying and classifying emotions in text data so that appropriate music recommendations may be made. The technology seeks to offer users a personalized music listening experience that corresponds with their overall emotional composition by analyzing the emotions expressed in tweets and linking them to music recommendations, potentially boosting their music discovery and enjoyment. The initiative might be helpful for users who want to get music recommendations curated for their personality as inferred from their tweets and it might provide information about the relationships between emotions and musical preferences.

CHAPTER 3

LITERATURE REVIEW

3.1 A qualitative and quantitative comparison between Web scraping and API methods for Twitter credibility analysis [1]

3.1.1 Introduction

Twitter is a very commonly used social network and a great source of information helpful for people and companies however in some cases it may not be reliable thus the existing studies focus on how credible the information shared on Twitter is. This paper focuses on two major data extraction techniques and performs a comparative analysis on both these techniques. The study's results highlight the disparities in efficiency and accuracy between both methods and also take into consideration other factors like time taken to respond based on the distance.

3.1.2 Characteristics and Implementation

A model is introduced to calculate how credible a tweet is, which involves text credibility through filters such as spam, bad words and misspellings, user credibility based on the account creation and verification and social credibility based on the followers. A framework that consists of a front-end and a back-end and conducts the credibility analysis in real-time is implemented as a Google Chrome extension. After output normalization, a quantitative and qualitative comparison of the two extraction methods is carried out. A quantitative comparison is done using the credibility model, requesting time for various locations, and execution time. A qualitative comparison is done using the attributes that are currently accessible and credibility metrics.

3.1.3 Components

Web scraper includes a script to access websites, parse and extract data, and store them in any format. It is more flexible, and faster, and a larger number of tweets can be retrieved however it is greatly affected by network speed and depends on the format of the webpage. Twitter API responds to the user's requests and is simple to use as provided by Twitter itself. It is independent of the webpage format and has greater attributes however the number of tweets obtained is less and has greater restrictions. Due to the recent changes in Twitter API Web Scraper is preferred.

3.1.4 Evaluation

By qualitative comparison of available attributes Web scraping retrieves, only 54.29% of the attributes considered, and credibility measures reveal that Twitter API can accomplish 60% of them while Web scraping can only perform 40% of them. Text, social media, and tweet credibility levels are equivalent for both strategies when compared quantitatively. Web scraping is 40 times faster than Twitter API in terms of processing time, and the performance of requesting time is found to be related to distance.

3.2 Forex Sentiment Analysis with Python [2]

3.2.1 Introduction

In this study, it was aimed to extract insights for forex market sentiment analysis by combining Twitter data and forex market price data. Two python libraries were utilized for tweet data retrieval which were Tweepy and Snscrape.

3.2.2 Characteristics and Implementation

Tweepy, as an official Twitter API, granted access to Twitter data through Python, requiring a minimum version of 3.7 while Snscrape, a versatile scraper tool, demanded a minimum Python version of 3.8. The choice of these libraries was driven by the need for a comprehensive approach in collecting diverse Twitter data, considering factors such as date range and extraction limitations.

3.2.3 Components

The data extraction process included a crucial step of data cleaning using Regular Expression techniques. Regular Expression, characterized by a series of characters defining search patterns, played a vital role in refining the extracted text data. Subsequently, sentiment analysis was performed using two libraries: VADER Sentiment from NLTK and TextBlob, offering negative, neutral, and positive scores, as well as polarity and subjectivity metrics.

3.2.4 Evaluation

Sentiment analysis of major forex pairs, including GBP/USD, EUR/USD, and AUD/USD, reveals a consistent dominance of neutral sentiment, often followed by positive sentiment. The absence of significant negative sentiment days suggests an overall positive bias. The visual correlation between positive sentiment and rising prices underscores the potential influence of social media sentiment on market trends, emphasizing the importance of integrating sentiment analysis with technical and fundamental analyses for informed decision-making in the forex market.

3.3 TwiScraper: A Collaborative Project to Enhance Twitter Data Collection [3]

3.3.1 Introduction

To overcome the limitations of Twitter API, a new module called Twi-FFN (Twitter Followers/Following Network) has been introduced for user-centered data collection. This module enables efficient Twitter data retrieval through scraping methods.

3.3.2 Characteristics and Implementation

This paper discussses the restrictions present in the Twitter API, and therefore offers an alternative method for collecting user networks more efficiently. By allowing parallel processing, Twi-FFN surpasses the limitations of the official API, making it a valuable tool for enhanced data collection. It is also discussed that the scraping method employed by Twi-FFN is legal, ensuring that the data obtained through this tool can be utilized within legal boundaries.

3.3.3 Components

The Twi-FFN module comprises several components that contribute to its effectiveness in Twitter data collection. It introduces a parallelized architecture for collecting user networks, overcoming the constraints imposed by the official Twitter API. The Twi-FFN2 module specifically focuses on retrieving the lists of followers and following for a given user, streamlining the process of constructing Followers Following Networks.

3.3.4 Evaluation

The module's parallel processing capability significantly enhances the speed and scalability of user network collection in comparison to the Twitter API.

3.4 Latent Personality Traits Assessment From Social Network Activity Using Contextual Language Embedding [4]

3.4.1 Introduction

Users reveal aspects of their personalities through the content they share with their social media followers and the trends in their interactions on online networking sites. The use of novel natural language processing methods for the analysis of social network activities is examined in this paper. The development of a lingo-stylistic personality traits assessment system uses the Myers-Briggs type indicator (MBTI) and big-five personality scales to analyze the personality traits of Twitter users based on their tweets. We propose a novel input representation mechanism that converts tweets into real-valued vectors using frequency, co-occurrence, and context (FCC) measures.

3.4.2 Characteristics and Implementation

The ensemble personality evaluation model evaluates how much a trait is manifested in a user's personality on a scale from 0 to 1. The model uses two different machine learning algorithms, linear Support Vector Machines and gradient-boosted decision trees as the combination of these two models yields more reliable and accurate predictions of personality traits. The proposed framework for classifying lingo-stylistic personality traits is a powerful method for analyzing a person's language use on Twitter to predict their personality traits. The framework is capable of accurately predicting a person's personality traits based on their language use on Twitter, which has many potential applications in fields such as psychology, marketing, and social media analytics.

3.4.3 Components

A technique for analyzing a person's language use on Twitter to predict their personality qualities is the proposed framework for identifying linguo-stylistic personality traits. After preprocessing, three techniques: count-based representation, Universal Sentence Encoder and GloVe word embedding are used to transform the text into real-valued vector representations. The Universal Sentence Encoder, a transformer-based sentence encoding model, utilizes the concept of "attention" to determine context-aware representations of words in a sentence that take into account both the ordering and identity of all other words. A smaller feature set was produced for the ensemble personality assessment model by genetically merging the two complementary representations.

3.4.4 Evaluation

It was assessed how well the TF-IDF, GloVe, and USE vector representations predicted personality traits. The outcomes demonstrated that GloVe and USE worked well together for the best performance. The outcomes demonstrated that the ensemble model outperformed both the gradient-boosted decision tree model and the linear SVM model. By adding noise to the dataset, the suggested framework's robustness was evaluated. The outcomes demonstrated that the architecture was noise-resistant, and even with noisy data, performance remained largely steady.

3.5 A Hybrid Deep Learning Technique for Personality Trait Classification From Text [5]

3.5.1 Introduction

The article explains how cognitive-based sentiment analysis has become popular for automatically spotting user behavior, particularly personality traits, from text posted on social media websites. Because the available methods are unsatisfactory, the authors suggest a hybrid deep learning-based model that successfully classifies 8 key personality traits using a convolutional neural network coupled with long short-term memory. Modern methods are compared to the proposed model, which has shown superior outcomes.

3.5.2 Characteristics and Implementation

To classify eight significant personality traits, the model combines LSTM and CNN. After preprocessing, word embeddings were used to translate the text into a numerical representation. After that, the suggested hybrid deep learning model for classification was fed the numerical representation. The model's recurrent portion uses LSTM to capture the temporal dependencies in the input text, while the convolutional portion uses a set of filters to extract features from the input text. The suggested model was developed using a benchmark dataset made up of Twitter messages. Eight personality traits were manually added to the dataset.

3.5.3 Features

Three modules make up the suggested approach for classifying personality traits from social media text: data collection, data pre-processing, and deep neural network implementation. A deep neural network is employed to transform the pre-processed social media reviews into a machine-readable format. The CNN model extracts essential components from the input text, whilst the LSTM model learns over time, in order to efficiently classify user ratings into several personality groups. Four categories are used to categorize the different personality types. Experimental studies show that the suggested technique successfully and accurately categorizes personality traits. By contrasting their suggested strategy with others, the authors show that their methodology outperforms other cutting-edge strategies.

3.5.4 Evaluation

The proposed model was trained on the benchmark dataset by the authors, who then assessed its performance using a number of measures including accuracy, precision, recall, and F1-score. The suggested CNN+LSTM model's performance was compared to those of machine learning classifiers that make use of the conventional Bag of Words feature representation method. The proposed CNN+LSTM model outperforms classic BoW representation-based machine learning classifiers like LR, RF, DT, SVM, KNN, and XGBoost as well as deep neural network techniques like Individual CNN, Individual LSTM, and Individual BILSTM.

3.6 A Survey of Sentiment Analysis from Social Media Data [6]

3.6.1 Introduction

The importance of sentiment analysis on social media to understand the emotions and opinions of users. Sentiment analysis is a method to infer the sentiment of a text and is gaining popularity in the business world due to technological advancements and increased access to social media. However, analyzing sentiments on social media can be difficult due to informal spellings, grammatical errors, and multiple languages. N-gram graphs can be used to evaluate substance-based sentiment analysis.

3.6.2 Characteristics and Implementation

The importance of analyzing social media comments to understand the emotions and opinions of users, which has become increasingly important in recent years is discussed. Two main methods of extracting sentiments are the lexicon-based approach and the classification-based approach. The process involves collecting all the corpus relating to a particular topic, standardizing the texts, entity recognition, identifying characteristics, sentiment revealing, and analyzing them.

3.6.3 Features

Lexicon-based approach involves evaluating valence shifters, which are responsible for communicating the attitude of the text organization, and then calculating the sentiment. The classification-based approach uses theories that suggest sentiments are independent of contexts or can be articulated through numbers. The Palavras software is used to determine the sentiments from Portuguese texts on social networks by using a hybrid unsupervised approach along with language processing, dictionary-based methods, and ontology procedures to create classifiers for emotions expressed on social media.

3.6.4 Evaluation

A study on sentiment analysis related to health shows how analyzing small text messages collected from Twitter could help understand people's emotions about respiratory tract infection A(HINI) vaccination. The accuracy of mining the sentiment was 84.29%, using Naïve Bayes classifier to identify optimistic and pessimistic tweets and maximum entropy classifier to identify unbiased and inappropriate tweets.

3.7 Employing BERT‑DCNN with sentic knowledge base for social media sentiment analysis

3.7.1 Introduction

Social media is a vital source for understanding public sentiment and enhancing local services. Opinions expressed on platforms like Twitter provide valuable insights in brief messages across various contexts. Recent research developments, such as sentiment analysis on social media data, allow businesses to better comprehend customer opinions, needs, and preferences by analyzing sentiments in user-generated content like tweets and posts.

3.7.2 Characteristics and Implementation

In order to produce more useful word representations, the research suggests a model for sentiment analysis on social media reviews that combines the BERT language model and DCNNs. For the purpose of creating word embeddings that are appropriate for this kind of data, the authors fine-tune BERT using a sizable dataset of social media reviews. The authors present a unique design that replaces the conventional sequential stacking of layers with three parallel dilated convolutional neural network layers and a global average pooling layer in order to increase the model's accuracy. This architecture makes it possible for the model to collect various context levels, improving sentiment analysis.

3.7.3 Components

BERT can learn contextual links between words in a sentence in both ways because to its bidirectional training approach. It is pre-trained using two training objectives, masked language modelling and next sentence prediction which enables it to learn how to understand the relationship between two sentences, as it is useful for down-stream tasks like natural language inference and question answering. The filter is applied to the input with holes or gaps between the filter weights in a dilated convolution. As a result, the convolution operation's stride is effectively increased, enabling the network to process more information over a wider area. The distance between the weights in the filter, which controls the degree of dilation, is called the dilation rate parameter. SenticNet is a well-known example of a Sentic Knowledge Base (SKB), which is a database that contains structured and organized information about concepts and their associated emotions, opinions, and sentiments.

3.7.4 Evaluation

The accuracy, precision, and recall for the proposed model were observed to be 87.1%, 0.87, and 0.86 respectively. This was compared with the evaluation metrics of some other models like LSTM, CNN, etc. and was found to be the most accurate.

3.8 Improving Sentiment Analysis for Social Media Applications Using an Ensemble Deep Learning Language Model [8]

3.8.1 Introduction

The deep learning approach for sentiment analysis was designed to eliminate manual feature engineering, demonstrating superior performance over other methods. Three main types of deep learning approaches—deep neural networks (DNNs), recurrent neural networks (RNNs), and convolutional neural networks (CNNs)—are employed for text and document classification.

3.8.2 Characteristics and Implementation

The proposed model is built upon the advanced FastText word embedding technique, which represents the feature space. This technique constructs word representations as a bag of character n-grams, alongside the word itself, enhancing the preservation of meaning for shorter words and capturing meaning for suffixes and prefixes. The feature vectors obtained from the word embedding model are then fed into LSTM networks in sequence, facilitating sentiment classification.

3.8.3 Components

The key components of the model include the utilization of the FastText word embedding technique and the implementation of LSTM networks. FastText represents words through character n-grams, contributing to the preservation of meaning. LSTM networks process the sequential input of feature vectors, enabling effective sentiment classification.

3.8.4 Evaluation

The model's performance and accuracy hinge on two crucial parameters: the total number of hidden neurons and the total number of hidden layers in the network. The study concluded that optimal classification performance, with an accuracy of 90.25%, was achieved by selecting two hidden layers and setting the number of neurons in the network to 200.

3.9 A Literature Survey on Sentiment Analysis Techniques involving Social Media and Online Platforms [9]

3.9.1 Introduction

Sentiment Analysis is a valuable tool for assessing the overall mood of a sample group through their textual expressions on platforms like social media and web reviews. It is discussed that sentiment classification techniques are categorized into machine learning, lexicon-based, and hybrid approaches. The machine learning approach, including supervised and unsupervised techniques, employs algorithms such as decision tree classifiers, neural networks, and Naïve Bayes.

3.9.2 Characteristics and Implementation

The lexicon-based approach, further subdivided into dictionary-based and corpus-based approaches, involves creating dictionaries based on opinion terms. In the former, a dictionary is initially formed, and online resources like dictionaries, thesauri, or WordNet are employed to expand it with synonyms and antonyms. In the latter, dictionaries are generated through seeds of opinion terms, growing through statistical or semantic procedures.

3.9.3 Features

The hybrid approach combines machine learning and lexicon-based strategies. It often employs improved Naïve Bayes and SVM algorithms together, incorporating feature selections like unigrams and bigrams to mitigate the gap between positive and negative sentiments. Studies indicate that this hybrid approach significantly enhances sentiment classification accuracy.

3.9.4 Evaluation

To assess the effectiveness of sentiment analysis techniques, evaluation becomes crucial. Different approaches, including machine learning, lexicon-based, and hybrid methods, are subjected to rigorous evaluation. Notably, the combination of machine learning and dictionary-based methods has been shown to yield substantial improvements in sentiment classification accuracy.

3.10 An emotion-aware music recommender system: bridging the User's Interaction and Music Recommendation [10]

3.10.1 Introduction

By adding emotional information into the recommendation process, this method aims to close the gap between user engagement and music suggestion. The two key components of the proposed system are emotion recognition and recommendation building. The emotion recognition module examines a user's emotion after examining their pattern of input device usage. music and emotional data are combined in the suggestion generating module to produce personalized recommendations.

3.10.2 Characteristics and Implementation

An interaction vector with four elements was developed: the quantity of mouse clicks, the quantity of keystrokes, the duration of time taken to hold each keystroke, and the average time each mouse button is pressed. Additionally, the user's interest in that song is determined as a number between 1 and 5, if the user clicks on the song and navigates to the music details page. One additional point is added if he listens to the song online, and three more points are added if the music is downloaded. The interaction vector is given additional weight in this study using the exponential moving average approach, which improves the accuracy of forecasting a user's emotional state. The recommendation is made using collaborative filtering, which calculates how similar the users are to one another, and suggests music the user hasn't listened to earlier.

3.10.3 Features

Users' keyboard and mouse interactions are used to derive four important features. These patterns are based on user inputs, mouse clicks, and time spent on them. These are utilized as the foundation for the creation of EMAu, i vector. The first half of the system predicts the user's emotional state, and the recommendation section creates a custom playlist for the user based on how closely they resemble other users. This paper's key benefit is that it recommends music by implicitly mapping the user's emotion to the music instead of mapping the user's emotion and passing that emotion as input to a recommendation which may affect negatively if the emotion is classified incorrectly.

3.10.4 Evaluation

This paper used the RMSE measure to see how the interaction vector is weighted and chosen.They considered a different number of records and found that 240 records and an alpha value of 0.8 yielded the lowest error value. The accuracy was computed for several values to see how the recommendation list size parameter, S, affected precision. S = 5 provides the highest accuracy value. The evaluations show that the suggested strategy is highly accurate in identifying customers. preferred music.

3.11 An Approach to Integrating Sentiment Analysis into Recommender Systems [11]

3.11.1 Introduction

In [11], the authors present an innovative recommendation approach that combines sentiment analysis with collaborative filtering methods. The proposed recommender system is built on an adaptive architecture, incorporating advanced techniques for feature extraction and deep learning models centered around sentiment analysis. The integration of sentiment analysis becomes particularly crucial when dealing with sparse ratings data, as it can significantly enhance the quality of recommendations.

3.11.2 Characteristics and Implementation

The recommender system relies on an adaptive architecture that incorporates improved feature extraction techniques and hybrid deep learning methods. Notably, the use of sentiment analysis is emphasized, leveraging the advantages of BERT for feature extraction. The deep learning models applied for sentiment analysis include CNN LSTM and LSTM followed by CNN, with ReLU stacked on top of the classifier. This integration of deep learning models is expected to enhance the accuracy of the recommendation system.

3.11.3 Components

The proposed recommendation method adopts a user-based collaborative filtering approach, taking into account both explicit ratings and sentiment analysis extracted from users' reviews. The collaborative filtering methods evaluated include Singular Value Decomposition (SVD), Non-Negative Matrix Factorization (NMF), and SVD++ (a derivative of SVD). The incorporation of sentiment analysis aims to capture user preferences and sentiments, providing a more comprehensive basis for generating personalized recommendations.

3.11.4 Evaluation

[11] implemented user-based collaborative filtering with SVD, NMF, and SVD++ recommendation methods.To validate the sentiment classification models, accuracy, F-score, and AUC are used.For evaluating the recommendation method, they considered rating prediction and top-N recommendation. In the former case, metrics such as RMSE, MAE, and NMAE were utilized, while in the latter case, we employed MRR, MAP, and NDCG.The introduction of sentiment-based proposals significantly enhanced recommendation reliability when compared to traditional rating-based methods across both datasets.

3.12 Sentiment Analysis of Social Media Content for Music Recommendation [12]

3.12.1 Introduction

This paper adopts a content filtering strategy to capture an individual's current or very recent emotional state. The methodology involves sentiment analysis on users' tweets, leveraging both text and emojis to identify predefined emotional categories, ultimately predicting user interests based on their emotional context. The specific emotions considered in this work include happiness, fear, anger, disgust, and sadness.

3.12.2 Characteristics and Implementation

This tweet sentiment analysis method matches users' expressions, including emojis, with predefined variables using Naive Bayes and support vector machines. The algorithm, trained on a dataset of tweets, classifies live tweets into emotional categories, captured in real-time via the Twitter API. A recommender system categorizes songs based on the emotional classes identified, recommending the top 5 songs to users based on their emotional context.

3.12.3 Components

The key component is the sentiment analysis process, the incorporation of predefined emotional categories, and the use of Naive Bayes and support vector machines as classification algorithms and then recommendation of songs based on the emotion classified.

3.12.4 Evaluation

The system's prediction accuracy was assessed by inputting random tweets, and the results received subjective approval from users.

CHAPTER 4

PROJECT REQUIREMENTS SPECIFICATION

4.1 Product Perspective

A music recommendation system using sentiment analysis on Twitter data involves analyzing the emotional tone of tweets and using this information to suggest songs that match the user's personality and mood. By leveraging social media data, this system can provide a list of music recommendations based on the user's personality or current emotional state. However, simply recommending music based on a user's listening history was not always enough. Users often listen to music to match their current mood and their music taste largely varies based on their personal tastes. To address this, we are building a music recommendation system using sentiment analysis of a user’s social media.

4.1.1 Product Features

- Twitter Data Collection: The product collects data from Twitter and analyzes different tweets of the specified user..
- Emotional Classification: The product uses machine learning algorithms to analyze the emotional tone of tweets, determining whether they are positive, negative, or neutral.
- Music Recommendations: Based on the sentiment analysis of tweets, the product recommends songs that match the emotional tone of the tweets.
- Personalization: Personalization in music recommendation using sentiment analysis can enhance the user experience and increase engagement with music streaming services by providing personalized music recommendations that align with a person's emotional state.

4.1.2 User Classes and Characteristics

Various user classes that may use this product include:

- Music Enthusiasts: This user class is likely to use the system to discover new music and get personalized recommendations based on their emotional state. They may be avid music listeners and use the system to explore new genres or artists that align with their preferences.
- Mood-based listeners: This user class is likely to utilize the system to find music that matches their mood or emotional state. They may utilize the system to find music that calms them down when they are feeling anxious or music that pumps them up when they are feeling low.
- Music Event Organizers: This user class is likely to utilize the system to curate playlists and recommend music for events. They may utilize the system to recommend music that matches the event's theme or helps create a particular ambiance.
- Music therapists: This user class is likely to use the system to recommend music for therapeutic purposes based on the emotional state of their clients. They may use the system to recommend music that helps their clients relax, focus, or manage their emotions.

4.1.3 Operating Environment

- Hardware platform: 8 GB RAM, above i4 processor
- Operating system and versions: Windows 10, macOS
- Software platform: Python, Flask, any web browser

4.1.4 General Constraints, Assumptions and Dependencies

- Regulatory Policies: Music recommendation systems need to comply with various regulations, such as copyright laws and data privacy regulations. These policies may limit the types of data that can be used to train the sentiment analysis models or the sources of music data that can be used for recommendations.

- Hardware Limitations: The performance of sentiment analysis models and the speed of music recommendation algorithms can be limited by the hardware capabilities of the system. If real-time music recommendations are required, the system may need to meet certain signal timing requirements, which can limit the hardware choices available to the developers.
- Limitations of Simulation Programs: Simulation programs are used to test the performance of the sentiment analysis models and music recommendation algorithms. However, these programs may not accurately simulate real-world scenarios, which can limit the choices available to developers in terms of testing and refining the system.
- Interfaces with Other Applications: It needs to interface with other applications, like social media sites, to access user data or to produce recommendations. These interfaces may be limited by integration options provided by these applications, which can impact the design and functionality of the music recommendation system.
- Criticality of Application: The criticality of the system can also limit the choices available to developers. If the system is used in safety-critical applications such as healthcare, certain performance or reliability requirements may need to be met, which can impact the design and implementation of the system.
- Safety and Security Considerations: The system needs to protect user data or prevent the recommendation of inappropriate or offensive content. These considerations can limit the types of data that can be used for training or the sources of music data that can be used for recommendations.

4.1.5 Risks

There are several risks associated with resource requirements and functionality in a music recommendation system that uses sentiment analysis on tweets:

- Scalability: The system may not be able to handle large volumes of data, leading to slower response times or even system failures.
- Accuracy: Sentiment analysis is not 100% accurate, and incorrect analysis could lead to inaccurate recommendations. The system must continuously monitor and improve the accuracy of its analysis.
- Bias: The sentiment analysis algorithm may have biases due to the training data used, leading to recommendations that are skewed toward certain demographics or genres of music.

4.2 Functional Requirements

- Validity Tests on Inputs: The system should perform validity tests on the input data obtained from Twitter, including checks for missing data, invalid characters, and spam or irrelevant tweets. The system should also ensure that the input data meets any necessary formatting requirements.
- Sequence of Operations: The system should be designed to perform the required operations in the correct sequence to generate accurate and timely music recommendations. The sentiment analysis should be performed before user and music classification to ensure that the recommended music matches the sentiment of the tweets.
- Error Handling and Recovery: The system must have robust error handling and recovery mechanisms to handle any unexpected errors or failures.
- Consequences of Parameters: The system should consider the consequences of the parameters used in the sentiment analysis and music recommendation algorithms.
- Relationship of Outputs to Inputs: The system should ensure that the music recommendations generated are relevant and consistent with the sentiment of the tweets. The relationship between the inputs and the outputs should be transparent and understandable to the user, to build trust and confidence in the system.

4.3 External Interface Requirements

4.3.1 User Interfaces

- Required screen formats with GUI standards: The screen format should be simple and straightforward. The interface should be designed using standard GUI styles and will include features such as buttons and text fields.
- Screen layout and standard functions: The screen layout should be organized and easy to navigate. Standard functions such as help should be provided to assist users in using the system effectively.
- Relative timing of inputs and outputs: The system will be responsive and provide timely outputs to user inputs. The time taken to process user inputs and provide recommendations should be minimal to ensure a positive user experience.
- Availability of some form of programmable function key: Programmable function keys will be provided to allow users to customize the interface based on their preferences.
- Error messages: The system will provide concise and clear messages for errors so users can understand how to troubleshoot or debug. The error messages should be displayed in a prominent location on the screen to ensure users do not miss them.

4.3.2 Hardware Requirements

- Memory: Around 8 GB of RAM.
- Network: High-speed internet connection with a minimum bandwidth of 20 Mbps.
- Processor: A multi-core processor with a clock speed of 2.0 GHz or higher.

4.3.3 Software Requirements

For each required product the following shall be provided,

- The software is developed for all devices running on web servers on devices which are able to connect to the internet.
- The application makes use of some third -party resources, such as required data from Kaggle.
- The application also extracts the user’s social media (tweets) from their account.
- The application will also make use of a Database to handle the backend, where the data in relation to the music as well as sentiment analysis will be stored. The database will be an SQL database.
- Operating Systems on which this can be used are Windows 10 and above, macOS
- Tools and libraries used will include scraping tools like Snscrape and Tweepy, Apify, tensorflow, keras

4.4 Non-Functional Requirements

4.4.1 Performance Requirements

- Accuracy: The system should have a reasonable level of accuracy in recommending music based on the sentiment analysis of tweets. It should be able to accurately classify the preferences of the user based on their tweets and recommend appropriate music.
- Speed: The system should be able to recommend music quickly and efficiently.
- Scalability: The system should be scalable, i.e, be able to handle a large volume of tweets and users. As the number of users and tweets increases, the system should be able to handle the load without any decrease in performance.
- Robustness: The system should be able to handle unexpected inputs, errors, and exceptions gracefully. It should be able to recover from errors and continue functioning without any significant impact on performance.

4.4.2 Safety Requirements

- Ethical considerations: The system should ensure that the recommendations made are ethical and do not promote offensive or discriminatory content.
- User safety: The system should ensure the user's safety and prevent the spread of harmful malware or inappropriate content. This can include using secure data storage practices and filtering out offensive language.
- User consent: The system should obtain user consent before capturing and analyzing their tweets. Users should be informed about the purpose of the system and the data that will be collected and used.

4.4.3 Security Requirements

- Data privacy: The system should ensure that user data is kept private and confidential. It should not share or expose user data to unauthorized persons or entities.
- Authentication: The system should require user authentication to prevent unauthorized access. Users should be required to provide valid credentials, such as usernames and passwords, to access the system.
- Secure Data Storage: Store data in a secure manner to prevent data breaches and cyber attacks.

4.5 Other Requirements

- Monitoring and reporting: The system should be monitored regularly for any security breaches or safety concerns. If any safety concerns are identified, they should be reported immediately to relevant stakeholders.
- Portability: We are building a web application so it can be accessed anywhere on any device connected to the internet.

CHAPTER 5

SYSTEM DESIGN

5.1 Overview

The system includes a web scraper to gather data from social media platforms, and a pre-processing step to clean and transform the data for classification of emotions which is then used in the recommendation algorithms. This project utilizes Content-Based Filtering with TF-IDF, GLoVe, USE, and DCNN to recommend music based on the emotion, providing a personalized and engaging user experience. The system utilizes both collaborative and content-based filtering techniques to provide music that is personalized to users based on their listening history, ratings, and emotions associated with the music they listen to.

The design incorporates a modular architecture with separate components for each step in the recommendation process. The system is designed to be scalable and able to handle large volumes of data efficiently. This document outlines the design and implementation details for each component, along with the data flow and relationships between components.

5.2 Design Considerations

5.2.1 Design Goals

The primary goal of the music recommendation system is to help users discover new music based on their sentiment expressed in tweets. The system should provide a personalized and engaging experience that helps users find music that matches their mood and preferences. The system should be user-friendly, easily accessible, and prioritize user privacy and data security.

The newly proposed system provides a personalized music recommender system based on user sentiment analysis and feedback, which improves the accuracy and relevance of music recommendations, makes it more personalized, and helps users to discover new music that aligns with their preferences and moods. The system uses sentiment analysis and machine learning algorithms to provide accurate and personalized recommendations.

Quality of service characteristics:

- Availability: The music recommendation system should be designed in such a way that the downtime is minimized.
- Security and privacy: The system should prioritize user privacy and ensure that user data is stored securely and handled responsibly.
- Speed: The system should provide recommendations quickly and efficiently, with minimal lag or delay.
- Personalization: The system should provide music recommendations that are personalized based on the user's mood and preferences.
- User-friendliness: The system should be easy to use, with a simple and intuitive interface.
- Security: The system should ensure that user data is stored securely and handled responsibly.
- Maintainability and scalability: The system should be designed for easy maintenance and scalability, with robust architecture and efficient data management.

5.2.2 Architecture Choices

- Web scrapers can access more data than the Twitter API. While the Twitter API provides access to recent tweets, a web scraper can scrape older tweets, as well as additional data such as user profiles, follower lists, and more. Also, web scrapers allow greater flexibility and customizability in terms of the data that is collected. With a web scraper, you can tailor your data collection to specific needs and parameters, whereas the Twitter API provides a more limited set of options.

- GloVe is a word embedding technique that captures global co-occurrence statistics to create word representations. It is a powerful and widely used technique for generating word embeddings that perform well on many NLP tasks. USE can be used to calculate the similarity between sentences and the emotions with their patterns.
- DCNNs can automatically learn hierarchical representations of raw text data, which enables them to capture complex patterns in text and achieve outstanding performance on sentiment analysis tasks. They can be trained on huge datasets of labeled emotion data and can learn to recognise patterns in the data that are indicative of different emotions. This can then be used to predict the emotion of unseen data.
- Content-based recommendation systems that use ML algorithms to analyze the features of items provide several advantages over collaborative filtering. They do not require user data, which can be a privacy concern, and they do not suffer from the cold-start problem. They can also recommend items that the user might not have encountered, providing greater serendipity.

5.2.3 Constraints, Assumptions and Dependencies

Assumptions:

- The system assumes that users have public Twitter accounts and have tweeted quite a number of tweets to get a better understanding of their personality.
- The web scraper provides reliable and consistent access to web data.
- The emotional classifier is accurate and can correctly identify the sentiment of the scraped data.
- The music dataset used for content-based filtering is comprehensive and up-to-date.
- Users will provide honest and accurate feedback for collaborative filtering.

Constraints:

- The system is limited to the availability and quality of the data obtained from the web scraper and the music dataset.
- The performance of the emotional classifier and recommendation system may be affected by the size and complexity of the datasets.
- The user interface and functionality may be limited by the capabilities of the web application framework used for development.

Dependencies:

- The system relies on access to the web data obtained by the web scraper and the music dataset.
- The sentiment analysis algorithm and recommendation system depend on the quality and completeness of the datasets.
- The music that is recommended is dependent on the sentiment classified by the emotion classifier.

5.3 High-Level System Design

5.3.1 Logical user groups

- Registered Users: Users who have created an account and have access to all the features of the application such as rating songs, viewing recommendations, and accessing their user history.
- Administrators: Users who have administrative privileges and can manage the application, including managing user accounts and data, managing music data, and updating the application.

5.3.2 Application components

- Web scraper: responsible for scraping data from Twitter.
- Data processing and storage: responsible for processing the scraped data, storing it in a database, and preparing it for analysis.
- Emotion Classifier: This component will use NLP techniques and ML to classify tweets into different emotions.
- Content-based filtering algorithm: responsible for recommending music based off emotion and the features of a given song.
- Collaborative filtering algorithm: responsible for recommending music based off the listening history and ratings of users.
- User interface: responsible for displaying the recommendations to the user, allowing them to rate songs, and providing a search function to find specific songs or artists.

5.3.3 Data components

- User data: This would contain information about users such as their name, email, user ID, and user history.
- Tweet data: This would include data retrieved from Twitter such as tweet ID, tweet content, and the user who posted the tweet.
- Music data: This would contain data about the music such as track name, artist name, release year, album name, and genre.
- Rating data: This would include user ratings for various songs, such as the song ID, the user ID, and the rating given by the user.
- Training data: This would include the data used to train the ML models such as tf-idf, GloVe, USE, and DCNN.
- Recommendation data: This would be the data generated by the recommendation algorithm, which would contain the recommended songs and their associated scores.

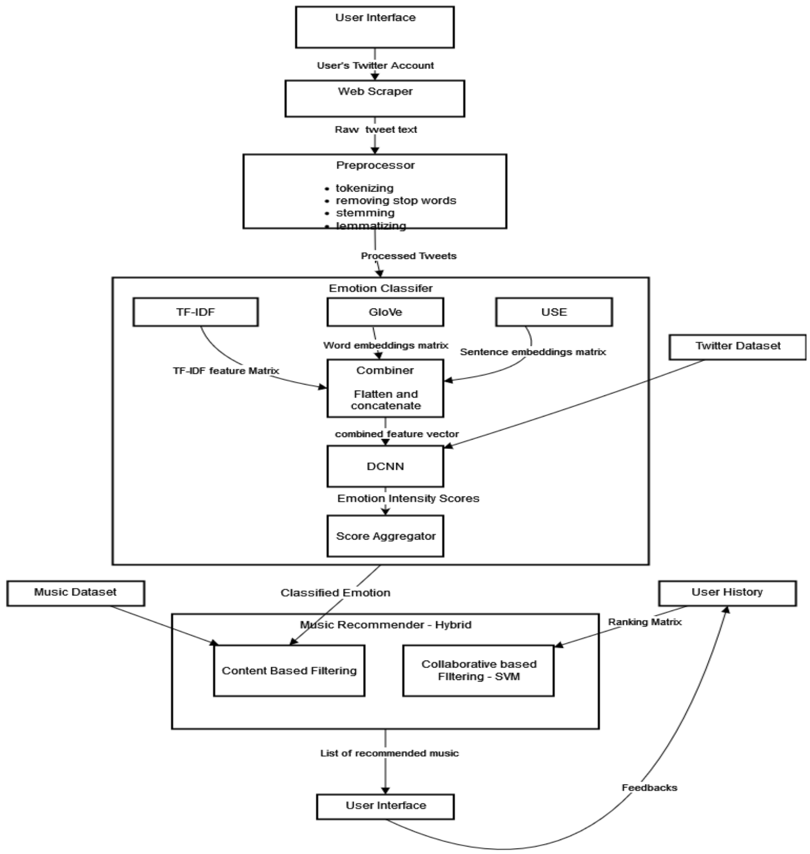

Figure 1. System Architecture Diagram

5.3.4 Module

Project Management:

- Defining project scope and objectives
- Identifying project stakeholders and their roles
- Developing a project plan that includes timelines, milestones, and resource requirements
- Assigning tasks to team members and monitoring progress
- Managing risks and issues that may arise during the project
- Conducting regular reviews and evaluations of project performance

Code organization:

- Defining coding standards and guidelines for the project
- Establishing a clear and consistent naming convention for functions, variables, and files
- Creating a modular code structure that separates different functionalities into modules or classes
- Maintaining a consistent file and directory structure for the project
- Ensuring that the code is well-documented, including comments in the code and a user manual or technical manual
- Conducting regular code reviews and evaluations to ensure that coding standards and guidelines are being followed

5.3.5 Security

- Authentication: Users should be required to log in with a username and password to allow access to the system after verification.
- Authorization: Once a user is authenticated, the system should have mechanisms in place to ensure they only have access to the parts of the system they are authorized to use.
- Encryption: Sensitive data like user credentials, ratings, and user histories should be encrypted to prevent unauthorized access.
- Access controls: It should be put in place to control access to the system's resources and data to only authorized users.
- Regular updates and maintenance: Regular updates and maintenance of the system can help to ensure that security vulnerabilities are addressed in a timely manner.

5.4 Design Description

5.4.1 Master Class Diagram

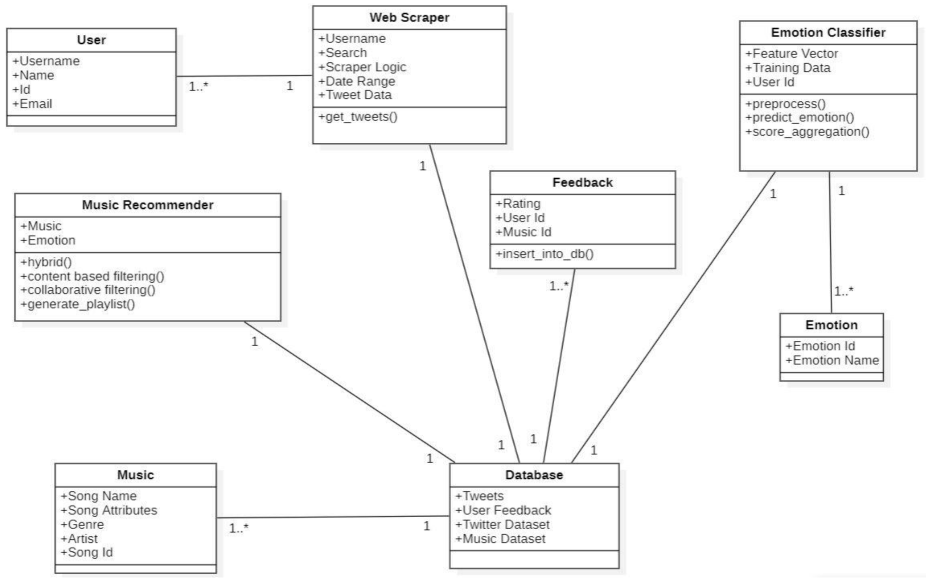

Figure 2. Master Class Diagram

5.4.2 ER Diagram

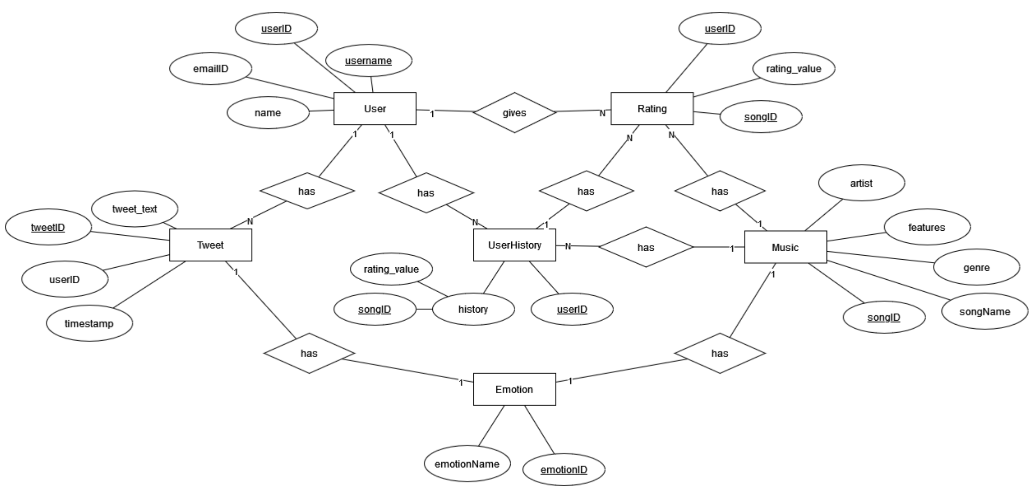

Figure 3. Entity - Relationship Diagram

5.4.3 User Interface

- Home Page: This is the first page that the user sees when they access the system. It can have options for signing up, logging in, and accessing various features of the system.
- Register Page: This page allows new users to create an account on the system. It can have fields for entering personal information such as name, email, and password.
- Login Page: This page allows existing users to log in to the system. It can have fields for entering the email and password associated with the account.
- Social Media Page: This page asks user to choose either twitter or threads to retrieve their data from and then asks them to enter their username on the respective social media.
- Music Recommendation Page: This page shows the music recommendations generated by the system. It can display information such as the song title, artist name, and a link to listen to the song.

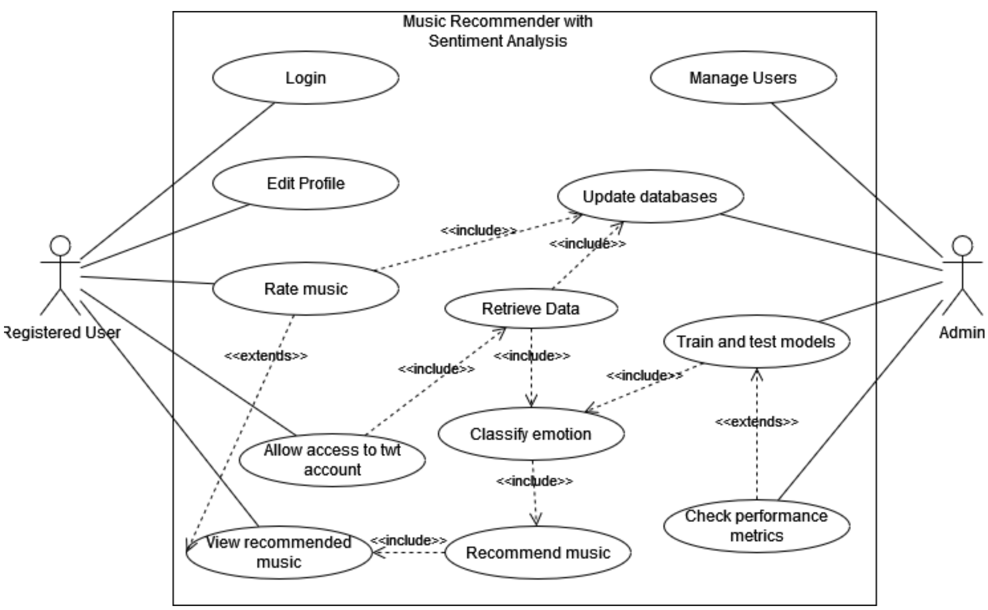

Figure 4. Use-Case Diagram

5.4.4 Sequence Diagram

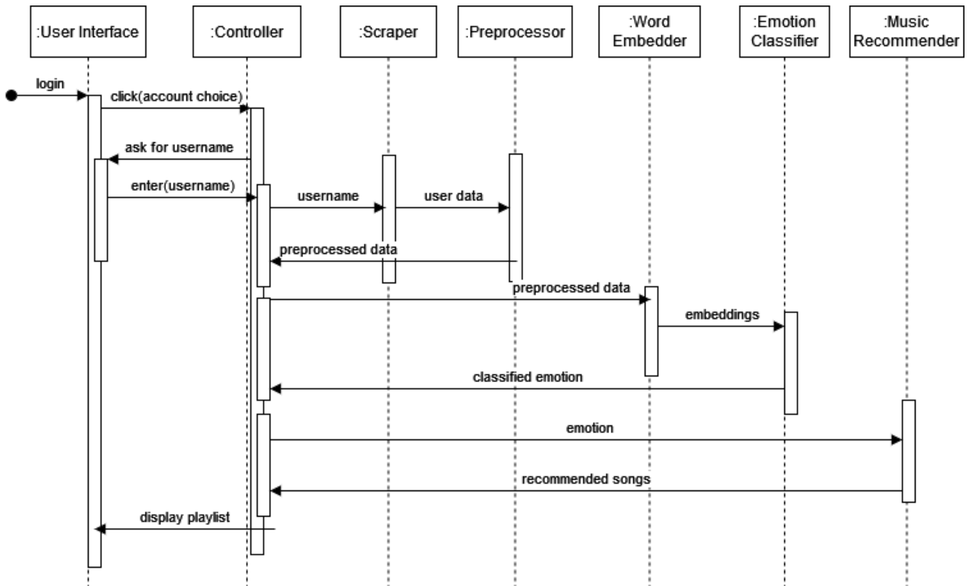

Figure 5. Sequence Diagram

5.5 Reusability Considerations

- The reusability considerations that can be taken into account to ensure that the project is easily maintainable and scalable are:
- Use of existing libraries and frameworks: To reduce development time and effort, existing libraries and frameworks can be utilized for tasks such as web scraping, sentiment analysis, and recommendation systems.
- Modular design: The project can be designed with modularity in mind, so that individual components can be easily reused in other projects. For example, the sentiment analysis module can be separated from the recommendation engine and reused in other sentiment analysis projects.

5.6 Help

- Required manuals will be made available to the users and admins to aid in the usage of the system including:
- The user manual: A manual that provides stepwise instructions on how to use the music recommendation system. It would include information on how to register for an account, how to search for music, how to rate music, and how to access recommended music based on the user's preferences.
- The technical manual: provides detailed information on how the system works and how it was developed. It would include technical specifications, such as the programming languages and frameworks used, as well as a description of the data sources and algorithms used for music recommendation. the system, as well as troubleshooting tips and suggestions for future development.

5.7 Design Details

A music recommender system with sentiment analysis will rely on a combination of platforms, systems, and processes to function properly. Some of the key areas:

- Web scraping: The system will rely on web scraping to extract tweets related to music. We can perform this using python libraries or web scraping tools.
- Emotional Classifier: The system will use an emotional classifier composed of TF-IDF, GloVE, USE and DCNN to determine the emotion of each tweet. The overall emotion of a user is then found by using an aggregator.
- Recommendation engine: The system will need a recommendation engine that can take into account the emotion of each user and suggest appropriate music.

- Databases: The system will require databases to store user's tweets, the twitter dataset for training and testing the emotional classifier, the music dataset to recommend music and to store user's history.

The effectiveness of the music recommendation system will rely on its ability to seamlessly integrate and balance the various platforms, systems, and processes involved. This must be done while keeping in mind critical factors such as novelty, innovativeness, interoperability, maintainability, and performance, all of which are crucial to the system's smooth operation and its ability to achieve its goals.

5.8 Low-Level Design

5.8.1 Use Case Description

Table 1. Use-case Items and their Descriptions

Use Case Item	Description
Registered User	The registered user is the person who enters their twitter/threads username to receive music recommendations.
Admin	The admin performs tasks such as managing users, updating the database, and overseeing the system's operations.
Login	This use case allows registered users to log into the system by providing their credentials
Edit Profile	This use case allows registered users to edit their profiles to update personal information,
Rate music	This use case enables registered users to provide feedback on music they have been recommended to improve the music recommendations.
Allow access to twitter account	This use case allows the web scraper to access the user's account so that tweets/threads can be extracted.

View recommended music	This use case enables registered users to view music recommendations generated by the system based on the emotion classification of their tweets.
Manage Users	This use case allows admins to manage user accounts, including tasks like creating, updating, or deleting user profiles.
Update Database	This use case represents the process of updating the system's database with user feedback.
Retrieve Data	This use case enables the user’s tweets/threads to be extracted through web scraping.
Classify emotion	This use case enables the system to analyze the emotion of a user's tweets/threads.
Recommend music	This use case enables the system to generate recommendations based on the emotion that is classified.
Train and test model	This use case represents the training and testing of machine learning models or algorithms used for music recommendations and sentiment analysis.
Check performance metrics	This use case enables the system to evaluate the performance of its recommendation and sentiment analysis algorithms. This involves tracking metrics like accuracy, precision, recall, and F1-score to ensure the system's effectiveness.

5.8.2 Class Description

5.8.2.1 User Class

Data members:

Data Type	Data Name	Access Modifiers	Initial Value	Description
String	Username	public	“”	stores username
String	name	public	“”	stores name
integer	Id	public	0	stores unique id
String	Email	public	“”	stores email id

5.8.2.2 Web Scraper Class

Data members:

Data Type	Data Name	Access Modifiers	Initial Value	Description
String	Username	public	“”	stores username
String	Search	public	“”	stores search data
date	Date Range	public	01-01-2000	stores start date from which tweets are to be retrieved
String	Tweet data	public	“”	stores user’s data

Data methods:

get_tweets(): This method retrieves a user's tweets from their Twitter account using a specified username and date range as input parameters. It produces a .csv file with the user's tweets as output, but exceptions may arise if the account is private or non-existent.

get_threads(): This method retrieves a user's threads from their Threads account using a specified username as input parameters. It is similar to get_tweets().

5.8.2.3 Emotion Classifier Class

- Data members:

Data Type	Data Name	Access Modifiers	Initial Value	Description
list	Feature_vector	public	[]	stores the combined feature vector of USE, GLoVe and Tf-idf
String	Training_data	public	“”	stores training data
Integer	userid	public	0	stores unique userid

Data methods:

preprocess(): This method preprocesses user tweets or threads stored in a specified .csv file, which is taken as input and produces a cleaned .csv file of the user's tweets/threads. Exceptions may arise if no tweets/threads are retrieved.

predict_emotion(): This method predicts emotions for each entry in a specified .csv file, taking the cleaned data from the input file and producing an output file with classified emotions for each user entry. Exceptions may occur if the input .csv file contains zero tweets/threads.

score_aggregation(): This method aggregates the classified emotions from the user's .csv file, determining the overall emotion. Taking each tweet classified with an emotion, as input, it produces an overall emotion.

5.8.2.4 Emotion Class

Data members:

Data Type	Data Name	Access Modifiers	Initial Value	Description
integer	emotion_id	public	0	stores unique id of each emotion
String	emotion_name	public	“”	stores name of each emotion

5.8.2.5 Music Recommender Class

The music recommender class is used to define the music recommender module of the system. This is used to recommend music to the user based on the classified emotion and user feedback.

Data members:

Data Type	Data Name	Access Modifiers	Initial Value	Description
list	music	public	[]	stores the music that can be recommended based on the emotion
integer	emotion	public	“”	stores id of the overall emotion classified

Data Methods:

content_based_filtering(): This method analyzes song attributes and suggests items similar to the user's expressed interests, leveraging the classified emotion from the tweet or thread as input. It gives a list of recommended songs as output. Exceptions may occur if there are issues with incomplete or incorrect attribute data.

*collaborative_filtering():*This method recommends songs based on user feedback and the classified emotion from the tweet or thread. It takes the emotion as input, providing a list of suggested songs.

hybrid(): This method combines the two recommender systems, utilizing the classified emotion from the tweet or thread as input to offer a final list of music recommendations as output.

generate_playlist(): This method is used to generate a playlist for the user which includes all the music recommendations that were given.

5.8.2.6 Music Class

Data members:

Data Type	Data Name	Access Modifiers	Initial Value	Description
String	song name	public	[]	stores the name of the song
List	song attributes	public	“”	stores the different attributes of songs like tempo, etc.
String	genre	public	“”	stores the genre of the song
String	artist	public	“”	stores the artist of the song
integer	songid	public	“”	stores the id of a particular song

5.8.2.7 Database Class

Data members:

Data Type	Data Name	Access Modifiers	Initial Value	Description
String	tweets/threads	public	[]	stores the tweets/threads of the user.
integer	user feedback	public	“”	stores the feedback that the user provides
-	twitter dataset	public	-	stores the twitter dataset that is used to train the sentiment analysis model.
-	music dataset	public	-	stores all the songs which are used to give recommendations to the user

5.8.2.8 Feedback Class

Data members:

Data Type	Data Name	Access Modifiers	Initial Value	Description
integer	rating	public	0	stores the rating of the user for the recommended music.
integer	userid	public	0	stores the userid of the user.
integer	music id	public	0	stores the id of a song

Data Methods:

insert_into_db(): This method inserts a given record into the database, taking the record as input. Upon execution, the record successfully gets added to the database. Exceptions may occur if the record is not properly inserted into the database due to a data type error, etc.

CHAPTER 6

PROPOSED METHODOLOGY

There are various processes involved in putting this concept into action. First, the dataset of tweets from Twitter users is prepared. The text data must be extracted, and cleaned up, and each tweet must be assigned to a certain emotion group. This includes preprocessing techniques like stop word removal, stemming, and lemmatization.

The preprocessed text data will next be subjected to natural language processing techniques, such as creating word embeddings by combining TF-IDF, GloVe, and USE. An emotion classifier model will use these embeddings as input to categorize each tweet into one of several emotion groups. A score aggregator is used to then find the overall emotion of the user from all the categorized tweets with more weightage placed on the user's recent tweets.

Four models were initially used to classify emotions which were LSTM, Bi-LSTM, CNN and DCNN. Three parallel layers of DCNN which have dilation rates 1,2 and 4 are stacked with a global average pooling layer to fine-tune the model, which showed the highest accuracy.

Table 2. Comparison of models used for Emotion Classification

Models	Accuracy
LSTM	38%
Bi - LSTM	10%
Convolutional Neural Network (CNN)	16%
Dilated Convolutional Neural Network (DCNN)	53%

From Table 2, we infer that DCNN demonstrated the highest accuracy among the models considered for emotion classification and therefore, it was selected as the final model while integrating word embeddings. The decision to incorporate word embeddings into the model was made to enhance its performance and capture more nuanced representations of emotions in the given context, and the accuracy further increased to around 80-85%.

Hence, the DCNN (Dilated Convolutional Neural Network) model is used for classifying the emotions expressed in textual data, enabling sentiment analysis, and understanding the emotional content of text. DCNN is deployed to forecast the emotion category of fresh, unread tweets once it has been trained on the labeled dataset. The new tweets are taken from the user's account with the help of a web scraper once access to it has been allowed by the user themself.

The user will then receive music recommendations based on their expected emotion category using a hybrid recommendation system. The content-based filtering algorithm is used to recommend music based on the classified emotion and the collaborative filtering part is used to introduce more personalization by using a database that stores user histories. Hence, a hybrid recommender system is used. SVM, Decision Tree Classifier, Random Forest and XGBoost were trained and tested on the dataset to check which performed the best.

Table 3. Comparison of the ML algorithms used for Content-based Recommendations

Models		**Accuracy**
Support Vector Machine	Linear Kernel	84.48
	RBF kernel	89.01
Decision Tree Classifier		90.57
Random Forest		94.04
Extreme Gradient Boosting (XG Boost)		95.63

Table 3 shows that the SVM with a linear kernel achieves an accuracy of 84.48%, while the SVM with an RBF kernel outperforms it, exhibiting greater accuracy. This superiority can be attributed to the latter's capacity to capture intricate patterns within the data, making it suitable for complex and high-dimensional datasets. This also makes SVM models computationally intensive. Decision trees are known for their computational efficiency but carry the risk of overfitting. Ensemble methods like Random Forest manage to strike a balance by combining multiple decision trees, offering flexibility and competitive accuracy. The extreme gradient boosting method, XGBoost, consistently outperforms Random Forest, predominantly due to its distributed computing capabilities and advanced feature engineering.

Table 5. Comparison of Evaluation Metrics for the ML algorithms used for Content-based Recommendations

Models	**Evaluation Metrics**			
	Precision	**Recall**	**F1 score**	**Support**
SVM	0.84	0.84	0.84	55588
Decision Tree	0.91	0.91	0.91	55588
Random Forest	0.94	0.94	0.94	55588
XG Boost	0.97	0.97	0.97	55588

It is observed from Table 5. that XGBoost and Random Forest have the highest precision, recall, and F1 scores indicating that they perform well in terms of classification accuracy on the given dataset.

CHAPTER 7

IMPLEMENTATION

7.1 Web Scraper

This component scrapes tweets from a Twitter account or threads from the threads account whose username is provided by the user.

7.1.1 Twitter Scraping

To initiate a Twitter session and log in, the target username for tweet retrieval is specified. Once logged in, tweets are collected from the specified account. This process involves accessing the user's profile details and extracting their tweets, providing a comprehensive view of their recent activities and shared content. This streamlined approach allows for efficient monitoring and analysis of the target user's Twitter presence, enabling a more in-depth understanding of their online interactions and contributions.

7.1.2 Threads Scraping

If the user chooses to use their threads account for analysis, the target username for threads is specified for threads retrieval.The user information is then fetched and the threads are collected from the user’s account.

7.2 Data Preprocessing

Applies preprocessing steps such as tokenization, stopword removal, and lemmatization to clean the text data which includes: conversion to lowercase, removal of URLs, hashtags, mentions, special characters, and numbers, tokenization and expansion of contractions, removal of stop words, punctuations, and lemmatization of tokens. The pre-processed tweets are then stored as a csv file for further use.

7.3 Word Embeddings

This combines multiple embedding techniques (GloVe, TF-IDF, USE) to represent tweets, allowing for a richer understanding of text data. The resulting embeddings are used in downstream tasks like sentiment analysis and music recommendation

7.3.1 GloVe Embeddings

The process involves initially loading pre-trained GloVe word embeddings into a dictionary. For every word within these preprocessed tweets, the corresponding GloVe word embedding is retrieved from the dictionary. The final step entails calculating the average word embeddings for each individual tweet, thereby generating a matrix that encapsulates the collective embedding representation of the entire set of tweets. This systematic approach converts the raw text data into a numerical format, facilitating subsequent analysis and insights based on the GloVe word embeddings.

7.3.2 TF-IDF Vectorization

A TF-IDF vectorizer is initialised. Following this, fit and transform the preprocessed tweets to compute TF-IDF vectors, capturing the significance of words within the dataset. Retrieve the feature names (words) corresponding to each column in the TF-IDF matrix to gain insight into the prominent terms. To enhance practical handling, convert the TF-IDF vectors to a dense array, streamlining subsequent operations. This methodical sequence enables the transformation of textual data into a numerical format, preserving the importance of terms through TF-IDF weighting and facilitating efficient analysis.

7.3.3 Universal Sentence Encoder

Initiate by loading the Universal Sentence Encoder model, a robust tool for capturing nuanced sentence semantics. Proceed to encode the given tweets using the Universal Sentence Encoder (USE), generating tweet embeddings that encapsulate contextual meaning. This streamlined process transforms the text data into numerical representations, preserving essential nuances for subsequent analyses or applications that demand a comprehensive understanding of the tweet content.

7.3.4 Combining Embeddings

The embeddings from GloVe, TF-IDF, and USE are combined along a new dimension, merging semantic and contextual information. Uniformize the arrays by padding with zeros to match the maximum number of tweets, ensuring consistent handling. Save the resultant stacked embeddings to a file, creating a comprehensive and standardized representation of the tweet dataset for future analysis or model training.

7.4 Emotion Classifier

The test and train datasets containing text and emotion labels are loaded along with the preloaded embedding files corresponding to the test and train datasets, leveraging embeddings for each tweet and emotion labels from the dataset. The input layer is defined to accommodate variable-size inputs. Incorporate dilated convolutional layers with varying dilation rates (1, 2, 4), each followed by batch normalization and global average pooling. The outputs are concatenated along the last axis to capture information at diverse receptive field sizes. Dropout layers are introduced to prevent overfitting, followed by a dense layer applying ReLU activation and kernel regularization. The final output layer is defined, utilizing the sigmoid activation function to produce probabilities for each emotion class. The model is compiled using the AdamW optimizer and binary cross-entropy loss. The model is trained on the training data, implementing early stopping mechanisms to curb overfitting. It is then evaluated on the test data, calculating the resulting test loss and accuracy. This systematic approach ensures robust model training, evaluation, and performance assessment for text classification based on emotion labels.

7.5 Hybrid Recommender system

7.5.1 Content based recommender system

Utilizing the Extreme Gradient Boosting (XGBoost) algorithm, classify music based on emotions for recommendation purposes. The dataset, containing song URLs, is processed through the Spotify API to convert URLs into song titles. This approach combines efficient machine learning with data preprocessing, resulting in a recommendation system that suggests songs with similar emotional characteristics to those the user has shown interest in.

7.5.2 Collaborative recommender system

Designed to predict user interests or preferences, this system collects and analyzes information about user behaviors and preferences within a group. A dataset is crafted, featuring user names, emotions, song titles, and ratings. Operating based on user input of emotion, the system employs two recommendation approaches. Firstly, for user-user similarity, it calculates the likeness between users with the target emotion and suggests highly rated songs from similar users. Secondly, for item-item similarity, it gauges the resemblance between songs of the desired emotion and recommends similar ones. The ultimate recommendation is a fusion of both systems, offering a comprehensive and personalized song suggestion based on user-emotion preferences.

CHAPTER 9

RESULTS AND DISCUSSION

The implementation of the emotion classification and music recommendation system produced promising results, demonstrating effectiveness in capturing and leveraging user-generated content for personalized music suggestions. The emotion classification model, incorporating TF-IDF, GloVe, and USE embeddings, exhibited high accuracy in discerning nuanced emotional states from tweets. The DCNN, trained on this diverse dataset, aggregated these emotional cues to predict overall user sentiment reliably. The accuracy ranged from 80% to 85% following the incorporation of the combined word embeddings.

The content-based music recommendation system, utilizing XGBoost to predict the emotion of the song based on its attributes and taking confidence scores into account, to find and deliver five songs that are specifically suited to the given emotion. The amalgamation of content-based and collaborative filtering- SVD, provided a balanced system, leveraging user behavior and preferences alongside tailored recommendations based on expressed emotions. It uses item - item similarity to recommend the songs if a new user is using an application or user has not made an interaction for that emotion previously, hence effectively handling the cold start problem posed by new users. This hybrid approach improved recommendation accuracy, addressing limitations of singular methodologies. The real-world applicability is evident in the system's dynamic adaptation to users' evolving emotional states, integrating real-time scraping for social media content to enhance responsiveness and ensure relevant recommendations.

The Flask application underwent thorough testing, encompassing unit and integration tests to ensure robust functionality. Unit tests were employed to scrutinize individual components of the application, verifying that each function operated as intended. This granular testing approach facilitated the identification and resolution of any isolated issues within the Flask app.

Integration tests were pivotal in assessing the seamless interaction between different modules and components. These tests evaluated how well the various parts of the application collaborated and whether data flowed correctly between them. Additionally, user authorization testing was conducted to validate the security measures in place. This involved simulations of different user scenarios to confirm that the app appropriately handled access permissions and maintained data integrity.

Furthermore, rigorous connectivity testing was performed to ensure the proper linkage between the Flask app and the databases. This included tests for data retrieval, storage, and update operations to guarantee that the app seamlessly interacted with both user and music databases. These tests were essential for validating the end-to-end functionality of the application, assuring users of a secure, reliable, and efficient experience.

While the system exhibits considerable success, there are inherent limitations. The effectiveness of the recommendation system heavily relies on the quality and quantity of user-generated content, i.e. their data on their social media of choice - twitter or threads. In cases of sparse or inconsistent data, the system's performance may be compromised. Additionally, concerns related to user privacy and data security must be addressed rigorously, especially when scraping information from social media platforms. The interpretability of the machine learning models, particularly the DCNN and XGBoost, poses challenges. Understanding the underlying decision-making processes of these models is crucial for building trust with end-users. Future iterations could focus on incorporating explainable AI techniques to enhance model transparency and user understanding.

CHAPTER 10

CONCLUSION AND FUTURE WORK

In conclusion, the creation of the system for classifying emotions and recommending music represents a noteworthy advancement in improving user experience and customization within the digital content consumption domain. The system's approach of allowing users to input their social media usernames for scraping tweets and threads has proven effective in capturing expressions and sentiments. Scraping the user’s posts allows the system to capture a dynamic snapshot of the user's emotional landscape and hence their overall temperament, allowing for personalized and timely music recommendations. The meticulous consideration of security, documentation, and user support ensures the system's reliability and accessibility in real-world scenarios.

The incorporation of sophisticated natural language processing methods, including TF-IDF, GloVe, and USE, has proven that the system can identify and classify emotional subtleties in user-generated content, especially tweets. Combining these methods with a deep convolutional neural network (DCNN) has made it possible to determine users' temperament holistically, which has improved the quality and accuracy of music recommendations. Utilizing machine learning algorithms to comprehend user preferences based on musical criteria, the system's adaptability is demonstrated by the integration of XGBoost for content-based music recommendation. It is a stable, user-focused platform that not only takes into account users' temperament but also customizes music recommendations per them. The system is reliable and accessible in real-world circumstances since security, documentation, and user assistance have all been carefully considered. A more thorough and individualized music suggestion system is offered by the addition of collaborative filtering algorithms to content-based suggestions. The incorporation of a user feedback system greatly improves the comprehension of user preferences.

Although the results of the current implementation are encouraging, there is room for improvement in the system's functionality in the future. Enhancing the emotion categorization models could be one area of emphasis. The accuracy and granularity of emotion recognition could be increased by investigating sophisticated deep learning architectures and conducting ongoing training on a variety of datasets.

The mechanism for suggesting songs might also be expanded to include integration with bigger streaming services. Partnerships with well-known businesses would give consumers access to huge music libraries and a wider variety of tunes. To continuously improve and customize recommendations, this integration may incorporate user feedback from these platforms. Additionally, investigating the possibility of using context-aware elements like location and time of day could provide the recommendations with an additional degree of customization. By doing this, the system would be able to modify recommendations per the user's surroundings and the situation at hand, offering a more comprehensive and personalized music recommendation experience.

In conclusion, the trajectory for the future takes a comprehensive approach, taking into account both the technical aspects of improving the model and the user experience by working together and integrating with larger music platforms.

REFERENCES

[1] Irvin Dongo, Yudith Cardinale, Ana Aguilera "A qualitative and quantitative comparison between Web scraping and API methods for Twitter credibility analysis" in International Journal of Web Information Systems © Emerald Publishing Limited 1744-0084 DOI 10.1108/IJWIS-03-2021-0037

[2] Luciana Abednego, Cecilia E. Nugraheni, Ame Fedora "Forex Sentiment Analysis with Python" in International Journal of Advanced Research in Economics and Finance e-ISSN: 2682-812X | Vol. 4, No. 1, 46-55, 2022

[3] Didier Henry. "TwiScraper: A Collaborative Project to Enhance Twitter Data Collection" WSDM ’21: The Fourteenth ACM International Conference on Web Search and Data Mining, Mar 2021, Virtual Event, Israel. pp.886-889, ff10.1145/3437963.3441716ff. ffhal-03267915ff

[4] P. K. K. N. and M. L. Gavrilova, "Latent Personality Traits Assessment From Social Network Activity Using Contextual Language Embedding," in IEEE Transactions on Computational Social Systems, vol. 9, no. 2, pp. 638-649, April 2022, doi: 10.1109/TCSS.2021.3108810.

[5] H. Ahmad, M. U. Asghar, M. Z. Asghar, A. Khan and A. H. Mosavi, "A Hybrid Deep Learning Technique for Personality Trait Classification From Text," in IEEE Access, vol. 9, pp. 146214-146232, 2021, doi: 10.1109/ACCESS.2021.3121791.

[6] K. Chakraborty, S. Bhattacharyya, and R. Bag, "A Survey of Sentiment Analysis from Social Media Data," in IEEE Transactions on Computational Social Systems, vol. 7, no. 2, pp. 450-464, April 2020, doi: 10.1109/TCSS.2019.2956957.

[7] Praphula Kumar Jain, Waris Quamer, Vijayalakshmi Saravanan, and Rajendra Pamula, “Employing BERT‑DCNN with sentic knowledge base for social media sentiment analysis” in Journal of Ambient Intelligence and Humanized Computing, January 2022

[8] Ahmed Alsayat, “Improving Sentiment Analysis for Social Media Applications Using an Ensemble Deep Learning Language Model” in Arabian Journal for Science and Engineering (2022) 47:2499–2511

[9] Raktim Kumar Dey, Debabrata Sarddar, Indranil Sarkar, Rajesh Bose and Sandip Roy, “A Literature Survey on Sentiment Analysis Techniques involving Social Media and Online Platforms” in INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH, VOL 1, ISSUE 1, May 2020

[10] Saba Yousefian Jazi 1 & Marjan Kaedi1 & Afsaneh Fatemi1,”An emotion-aware music recommender system: bridging the user’s interaction and music recommendation” in Multimedia Tools and Applications(springer),15 January 2021

[11] Cach N. Dang , María N. Moreno-García and Fernando De la Prieta ”An Approach to Integrating Sentiment Analysis into Recommender Systems “.This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license ,23 August 2021

[12] Akuma, S., Obilikwu, P. and Ahar, E,”Sentiment Analysis of Social Media Content for Music Recommendation”,Nigeria Annals of Pure and Applied Sciences Vol. 4 Issue 1, 2021

APPENDIX A: DEFINITIONS, ACRONYMS AND ABBREVIATIONS

- API: Application Programming Interface - a type of software interface, offering a service to other pieces of software.
- BERT: Bidirectional Encoder Representations from Transformers- pre-trained natural language processing model
- CNN: Convolutional Neural Network - deep learning neural network sketched for processing structured arrays of data such as portrayals.
- DCNN: Dilated Convolution Neural Networks - deep learning models that use a convolution operation with increased gaps between the input feature maps to increase the receptive field without increasing the number of parameters
- GB: GigaBytes
- MB: MegaBytes
- GloVe: Global Vectors for Word Representation - an unsupervised learning algorithm for obtaining vector representations for words.
- LSTM: Long Short-Term Memory networks - a variety of recurrent neural networks that are capable of learning long-term dependencies.
- RAM: Random-access memory - where the data is stored that your computer processor needs to run your applications and open your files
- SQL: Structured Query Language - used to access and manipulate databases
- TF-IDF: Term Frequency-Inverse Document Frequency - a measure that can quantify the importance or relevance of string representations in a document amongst a collection of documents
- USE: Universal Sentence Encoder - encodes text into high dimensional vectors that can be used for text classification, semantic similarity, clustering, and other natural language task.