JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 1047 - 1057
https://publishoa.com
ISSN: 1309-3452

Box-Office Analytics and Movie Recommender System Using Machine Learning Algorithms

P. Vimala Manohara Ruth, Kavita Agrawal

Assistant Professor, Department of Computer Science and Engineering,Chaitanya Bharathi Institute of Technology, Gandipet,Hyderabad, 500075,India

ABSTRACT

The film industry has boosted up with hundreds of movies during the pandemic. The film makers have tough and challenging time to produce movies which the audience are willing to watch. People all around the globe are binge watching their favorite movies and web – series in their pastime. Filmmakers are trying to boost up the quantity and quality of their projects because of high competition they are facing. Filmmakers need help to identify the kind of genres that people are willing to watch and also the film watchers with personalized recommendations.

A common platform is built for both film makers and viewers for predicting box office success and data analytics for analyzing trends in audience's interests. The Film makers will be equipped with features such as predicting the box office success of their project using various parameters that the machine learning model has been trained on. They can understand and analyze the social media engagements and strategize accordingly. Also, the film makers can make use of the Location suggestion feature to get the suggested locations for canning various scenes of their movies / web – series. On the other hand, the general audience will be able to use recommendation systems based on parameters like genre, OTT Platform and many others. They can also vote for the genres they are willing to watch which would help the film makers in understanding the interests of the audience. In this paper, the model was tested using various machine learning algorithms such as linear regression, decision tree regression and gradient boosting regression. Gradient boosting regression has shown better results with r2 score of 0.8960.

Keywords: *Content Based Filtering, Collaborative Filtering, Box-Office Prediction, Social media Analytics, Film Recommendation*

1. INTRODUCTION

The film-makers are having a great competition to grab box-office success due to the high demand of entertainment required by viewers due to the pandemic. Film - makers are investing crores of rupees for making quality films. So, Box Office success is of utmost importance for the film-makers to recover their investment. On the other hand, the general audience is also overwhelmed with lots of content across various OTT platforms. So, there is a need for a recommender system that can provide the viewers with the best recommendations. The objective is to create an application, common for both Filmmakers and viewers. Filmmakers can use the predicted Box Office figure so as to get an idea of the optimal budget to be invested. The Filmmakers also can get help from social media platforms such as Twitter and Youtube by viewing analytics related to their film. This would help them in controlling production costs and also understand the preferences of viewers. Film viewers will get recommendations across various OTT platforms based on their preferred genres and OTT platform preferences. For film viewers, we will be building a recommender system that will suggest films across various OTT platforms to which they have subscribed. The Algorithms to be used here are Content - based filtering and Collaborative filtering.

Content based filtering [13] uses item features to recommend other items similar to what the user likes, based on their previous actions or explicit feedback. The model doesn't need any data about other users, since the recommendations are specific to this user [14]. The model can only make recommendations based on existing interests of the user. In other words, the model has limited ability to expand on the users' existing interests.

Collaborative filtering is used to address some of the limitations of content-based filtering. Collaborative filtering [15] uses similarities between users and items simultaneously to provide recommendations. This allows for serendipitous recommendations; that is, collaborative filtering models can recommend an item to user A based on the interests of a similar user B. To predict the Box Office figure for a film, regression algorithms will be used since a continuous value is to be predicted, not a discrete value.

Multiple Linear Regression algorithm shows a linear relationship between a dependent (y) and one or more independent (x) variables, hence called linear regression [16]. Since linear regression shows the linear relationship, which means it finds how the value of the dependent variable is changing according to the value of the independent variable. Multiple linear regression is an extension to simple linear regression. In this setup, the target value depends on more than one variable.

Decision tree regression [17] observes features of an object and trains a model in the structure of a tree to predict data in the future to produce meaningful continuous output.
Ensemble methods [18] is a machine learning technique that combines several base models in order to produce one optimal predictive model.
Gradient boosting regression is a type of ensemble model based on boosting. A new decision tree is trained at each step against the negative gradient of the residual error.

The Scope of this work is to address the difficulties faced by film - makers in analysing the data and trends related to their film, our system will help in understanding the craze for the film in viewers. The filmmakers will be able to predict the Box - Office revenue based on several parameters such as "genre", "actors", "actresses", "director" etc. Film - makers will also get to see the analytics related to their film in Social media platforms such as Twitter and YouTube. Our system will also help makers in providing them a list of locations that would be suitable for canning their project.
On the other hand, the viewers will be able to get recommended films across the OTT platforms based on their preferences. Viewers will get the recommended films based on the OTT platforms to which they have subscribed. Initially, as the number of users will be low, Content based filtering will be used where the film recommendations will depend solely on the similarity of the film overviews and not based on those of other viewers.

Since, this is a portal where both, film - makers and viewers coexist, this provides the filmmakers with an opportunity where they can also get an insight of the kind of films that the viewers would want to watch in future and provide these details to the film - makers.

2. LITERATUTE REVIEW

2.1 Introduction to the Problem Domain Terminology

Film: A motion picture or moving picture. A film is a visual art-form to communicate ideas, stories, perceptions and feelings through the use of moving images. These images are usually accompanied by sound.
Filmmaker: A producer or director of motion pictures, especially one working in all phases of production.
Box Office Revenue: It is the revenue generated from ticket sales including any taxes and other levies.
Social Media Analytics: Social media analytics is the practice of gathering data from social media websites and analyzing that data using social media analytics tools to make business decisions.
Social Media Engagements: Social media engagement measures the public shares, likes and comments for an online business' social media efforts.
Film Viewer: A person who is interested in and watches films.
OTT Platform: OTT stands for 'Over The Top' and refers to any streaming service that delivers content over the internet. The service is delivered over the top of another platform, hence the term.

2.2 Existing Solutions

There are very few websites that try predicting the box office revenue of a film. Most of them are like news articles or blogs where the author manually analyzes the data related to a film across the internet. The author then gives a rough estimate of the Box Office revenue that the film might end up grossing in the first weekend. These very few websites that try predicting the performance of the film are designed for Hollywood films. For Social Media Analytics, Google Trends is an existing generic solution that can be used to get information about the number of searches for the film in Google search engine. But there is no analytics about the sentiment on the film among the audience. Also, there is a need for filmmakers to know about the viewer's interest as they vary with changing times so as to help them produce films that the users want to watch. Movie recommender systems are present in OTT platforms such as Prime Video, Netflix etc. but are specific to their platform for obvious reasons. The viewer needs to browse through the recommendations of subscribed OTT platforms and decide the film to be watched. A recommender system for users that tries recommending films across the platforms would be helpful.

2.3 Related Works

2.3.1 Prediction of Movies Box Office Performance Using Social Media (Published in 2013) [2]

Approaches Used: K-Means Clustering, Decision Tree and Naive Bayes
Description:
In [2], Data collected from social media platforms like Twitter, YouTube was used for classifying the Box Office performance of a film. Here, the dataset was specific to Hollywood films. The approaches used were K-means Clustering, Decision tree and Naïve Bayes. Here, they have used social media data like the followers of the actors, actresses etc. in twitter. Initially, the data is unlabeled, so they used K-means to form 3 clusters for Hit, flop and average. They have appended the respective cluster to the dataset so that a labelled dataset is obtained that can be used for Supervised learning. So, they had used supervised learning algorithms of Decision Tree and Naïve Bayes and found that Decision Tree gave

better accuracy.

2.3.2 Predicting Movie Success Based on IMDB Data (Published in 2017) [3]

Approaches Used: Linear Regression, Logistic Regression and SVM Regressor

Description:

In [3], Data collected from IMDB was used to train various machine learning models. They have used nominal features like Actors, Director, Writer, Production-House, Genre and numeric features like Budget, IMDb Rating, IMDb Votes, No of Ratings for prediction. The approaches used were Linear regression, Logistic Regression and SVM regressor. In Linear regression and SVM regressor, the box office revenue was predicted, which is a continuous value. In Logistic regression, since the predicted value has to be discrete, they have divided the Box office revenue into bins. The model then tried predicting the bin, that is, the range in which the Box Office revenue would be. It was found that Linear regression gave better results when compared to other models.

2.3.3 Design of Recommender System using Content Based Filtering and Collaborative Filtering Technique: A Comparative Study (Published in 2020) [4]

Approaches Used: Content based filtering, Collaborative filtering

Description:

This paper is about using Content based filtering and Collaborative filtering on Movielens dataset and making observations. This dataset has the ratings given by various viewers for various films. The genres of the films are also present. In [4], they observed that when the data was sparse, that is when the number of users is low, content based filtering performed better. When the data is not sparse, that is when the number of users is high, Collaborative filtering performed better.

2.4 Tools / Technologies Used

2.4.1 Anaconda

Anaconda [8] distribution comes with 1,500 packages selected from PyPI as well as the conda package and virtual environment manager. It also includes a GUI, Anaconda Navigator, as a graphical alternative to the command line interface (CLI).

Conda [8] analyses the current environment including everything currently installed, and, together with any version limitations specified, works out how to install a compatible set of dependencies, and shows a warning if this cannot be done.

2.4.2 Jupyter Notebook

The Jupyter Notebook [9] is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more. The "notebook" [10] term can colloquially make reference to many different entities, mainly the Jupyter web application, Jupyter Python web server, or Jupyter document format depending on context. A Jupyter Notebook document is a JSON document, following a versioned schema, and containing an ordered list of input/output cells which can contain code, text, mathematics, plots and rich media, usually ending with the ".ipynb" extension.

2.4.3 Twitter API

The Twitter API [5] can be used to programmatically retrieve and analyze data, as well as engage with the conversation on Twitter. This API provides access to a variety of different resources including Tweets, Users, Trends, Media, Places.

2.4.4 YouTube API

The YouTube Application Programming Interface [6] allows developers to access video statistics and YouTube channels data via two types of calls, REST and XML-RPC. Google describes the YouTube API Resources as APIs and Tools that let you bring the YouTube experience to your webpage, application or device.

2.4.5 Machine Learning

Machine Learning [11] is the field of study that gives computers the capability to learn without being explicitly programmed. ML is one of the most exciting technologies that one would have ever come across. As it is evident from the name, it gives the computer that makes it more similar to humans: The ability to learn.

2.4.6 Natural Language Toolkit (NLTK)

NLTK [7] is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum. This toolkit is one of the most powerful NLP libraries which contains packages to make machines understand human language and reply to it with an appropriate response.

2.4.7 Visual Studio Code

Visual Studio Code [12] is a free source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.

3. DESIGN OF THE PROPOSED SYSTEM

3.1 Block Diagram

Fig 3.1 represents the various features involved in our system. The data being passed between the various blocks of the system are also shown. The entry to our system is the login page for the filmmakers and viewers. Based on successful authentication, the user will be directed to the respective Dashboard. The dashboard will display the features that can be accessed and used. For Filmmakers, we display 4 features named Box Office Prediction, Social Media Analytics, Location Suggestion and Viewer preferences. On the other hand, for the film viewers, there will be a Film Recommendation feature which will work based on the Genre preferences and OTT preferences steps which act as input for the feature. The output of all the features are then displayed on the screen. For the Box Office Prediction, the revenues predicted for different IMDB ratings are displayed graphically. For Social media analytics, there are two types of analytics shown, one is twitter analytics and the other is Youtube analytics. Again, for both these types, the analytics are shown as graphs to compare the analytics of the films provided in the input. For Location Suggestion, based on the inputs, some pleasing locations are shown with their pictures. The Viewer preferences is a simple representation of genre preferences the viewers have selected on their side. The Film viewers will see the film recommendations based on their input preferences. These will be shown as film cards with their title, picture and a link to watch the movie in their preferred platform.

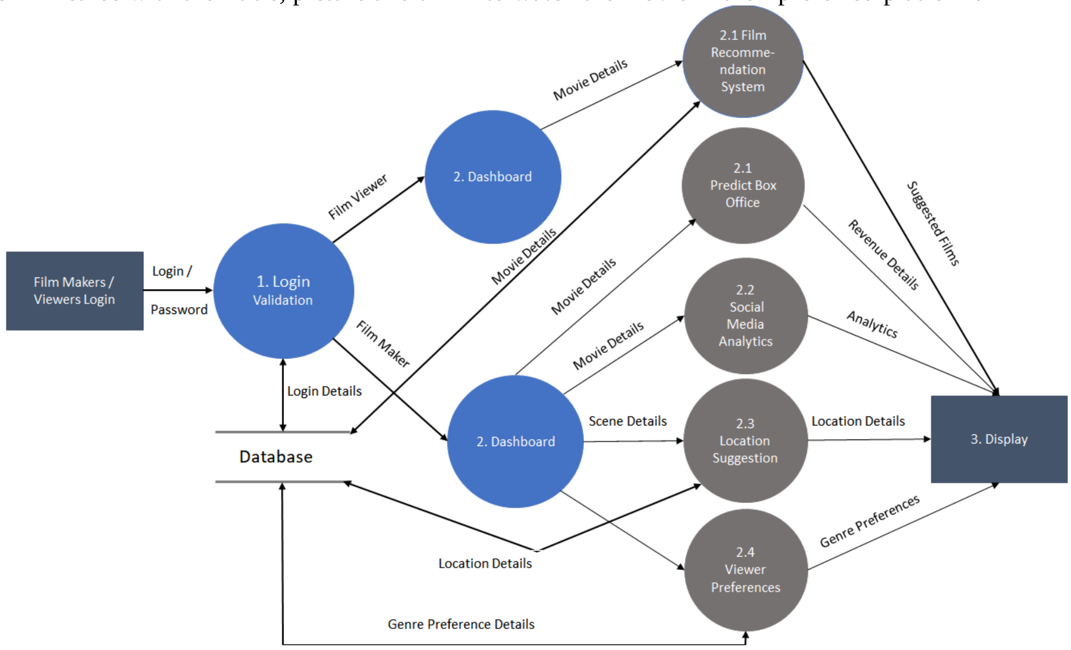


Fig 3.1. Block Diagram of the Proposed System

3.2 Module Description

This section of the report explains each module that is involved in our system. We have provided explanations for each of the modules included in both sides of our project, that is, the filmmaker and film viewer side of our application.

3.2.1 Box Office Prediction

For Box Office Prediction, we decided the independent features which would contribute to the dependent feature that is the Box Office revenue to be Actors, Actresses, Director, Genre, IMDB rating, Month of release, Year of release, Budget, Number of screens, Pan Indian release. Actors is a list of comma separated male protagonists who are part of the film. Similarly, actresses are also a list of comma separated female protagonists part of the film. Director is not a list but only a single valued attribute. Number of screens is an integer attribute which must be in the range of 500 - 15000. Budget of the film is also a positive integer in crores which signifies the money invested in the making of the film. Month of release is a drop-down list with all the months from January to December as options. Year of release is constrained to be in the range from 2013 to 2025. Genre is also a list of genres the film is made of. Pan Indian release is a binary attribute with Yes / No as values. The dataset is made of the above features along with the output feature which is the Box Office revenue generated in crores. The Gradient Boosting Regressor machine learning method will be used to predict the Box Office revenue for the input provided by the filmmakers for different IMDB ratings.

3.2.2 Social Media Analytics

Social Media analytics is another feature whose objective is to provide the filmmakers an insight of the engagements related to their film on social media platforms of Twitter and video platform of Youtube. So, twitter and Youtube analytics are the two types of analytics being provided. The input for the social media analytics feature is the base film, the films that this film must be compared with and the analytics type which can be either Twitter or Youtube. For Twitter analytics, three graphs will be displayed to the filmmaker. One graph is for representation of sentiment distribution of tweets of the base film. The other graph is to compare this sentiment distribution with those of other films. The last graph is to compare average sentiment scores of the films. For Youtube analytics, three graphs, that is, one each will be displayed for comparison of views, likes and dislikes of the films most watched videos on Youtube.

3.2.3 Location Suggestion

This feature is to help filmmakers with choosing locations for canning their songs or action scenes. This feature takes input as the scene type and budget range. Scene type has two values of songs and action. Budget range has been categorized into High, Medium and Low. The filmmaker shall choose the scene type for which they need locations and also the budget they wish to spend for that scene. Based on the input, few staggering locations will be displayed among which they can choose one which best suits their interest.

3.2.4 Viewer Preferences

This is a feature which will help filmmakers in understanding the genre belonging to which the viewers in our platform are willing to watch. This feature will have no input. The genre preferences of the film viewers in our platform will be used to display the graph between the genre and the number of interested viewers for that genre. The filmmakers can use this to know the changing trends and choices of films. Based on the genre preferences, the filmmaker can choose to make a film based on the most preferred genres. This would bring better hype and interest among the audience and hence would lead to greater Box Office revenue thereby increasing the probability of making profit out of the film.

3.2.5 Film Recommendation

This is a feature for film viewers using which they can receive recommendations personalied to their genre and OTT platform interests. We will take input for this feature as genre preferences and OTT preferences. Both these inputs can be multivalued, that is, the viewer can choose zero or more choices for genre and OTT platforms. Our algorithm will generate few recommendations for the viewer. At the output, the viewer will be able to see film titles, posters and a link that directs the viewer to watch that movie only on their preferred platform. This will help film viewers to not only obtain recommendations across all the OTT platforms but also to easily watch that film as we are eliminating the work of searching for the films in the platforms by providing the link to watch as well.

3.3 Theoretical Foundation

Ruth et.al 2021 [1] mentioned that XGBoost classifier has better performance because it has managed decision tree execution. This model is used to improve performance and speed of the model.
Box Office Prediction is a feature for prediction of revenue that might possibly be generated at the Box Office. Since this involves prediction and is not a deterministic problem that can be solved by using any specific formula, it is required to find patterns in the data to inference the future. So, to approach this problem of Box Office prediction, we have to use Machine learning.

Machine Learning involves a lot of algorithms and are broadly classified as Regression and Classification problems. Since, our problem is to predict revenue which is a continuous value, regression algorithms must be used. Among regression algorithms, we find a plethora of algorithms such as Linear regression, Polynomial regression, Support Vector Machine regression, Decision Tree regression, K nearest neighbors and also ensemble methods of Random Forest regression, Adaboost Regression, Gradient Boosting Regression. Based on our literature survey, we could find that for their dataset which is for Hollywood films, Linear regression and Decision Tree regression, Gradient Boosting regression proved to be better in terms of accuracy. For our dataset, Table 1 shows the r2 scores for different Machine Learning algorithms that we had tried out.

Table 1: R2 Scores of various algorithms

Algorithm	**R2 Score**
Random Forest Regression	0.7783
Decision Tree Regression	0.8116
Gradient Boosting Regression	0.8960

Linear regression is a supervised regression learning algorithm which predicts the value of a dependent variable based on values of the given independent variable. So, the linear regression algorithm finds out a linear relationship between the dependent and independent variables. Decision tree regression is another supervised regression algorithm that observes the features of a dataset and trains the model in tree structure for predicting continuous output data in the future. Boosting in machine learning is an ensemble method to use multiple simple models. Boosting is hence known as an additive model, because the weak models are added one at a time without making any changes in the existing tree models. Gradient boosting algorithm uses the method of gradient descent to minimize the loss. Decision trees are the models which are used as the weak learners in gradient boosting algorithm.

For film recommendation, there are approaches of Content based and Collaborative based filtering. Content based filtering algorithm generally suits well for systems with a smaller number of users. Collaborative based filtering algorithms tend to work well in systems with larger numbers of users. Content based filtering algorithm uses the item's feature vectors to find out similarity between the items and then, the top few most similar items are recommended. On the other hand, Collaborative filtering System does not use features of the items for recommendation. It uses the reactions of other users to recommend to a particular user. So, it recommends items based on similarity of users rather than using the similarity of items.

4. IMPLEMENTATION

We planned the design of our system to contain each feature as a separate module in our Django application. Hence, we designed our system to have a module for Box Office Prediction, Social Media Analytics, Location Suggestion, Viewer's interests and Film Recommendation modules. We decided to use PostgreSQL as a database to store all the data required by the modules for their functioning.

We used HTML, CSS, Javascript from front end development and Django, a python based framework for backend. We used PostgreSQL, an open source relational database management system for data storage and retrieval.

The first feature to be implemented is the authentication of users. We developed the front end for the login and registration pages. We created a database table to store user details. The backend logic was then implemented for proper authentication of the users before being able to use their accounts. The logout feature was later implemented.

For Box Office Prediction, we used a dataset containing the chosen features for training the Gradient boosting Regressor model. We tried out various regression models of Decision Tree Regressor, Random Forest Regressor but could not achieve the accuracy and consistency being obtained by the Gradient Boosting Regressor model. The machine learning model has been saved in a file and this file has been used in the BoxOfficePrediction module of our Django application. The backend logic required has been implemented in this module to do all the steps of data processing, prediction and also displaying the output. Since the output is graphical, we used Chart.js which is a Javascript based library to generate graphs for our data. We tested if the feature is working correctly by giving various inputs. We fixed the defects in our system that we had found during this phase.

Then, we started with implementation of the next feature, that is, Social Media Analytics. We implemented the backend logic where we used Twitter and Youtube API's for getting the required data and did some data processing for analysis. After implementation of the backend for our analytics feature, we implemented the frontend for this feature and connected it with the frontend to display the output graphs. For this feature, we tested different scenarios after implementation to find issues.

We started with the Location suggestion feature next which doesn't contain much backend logic to be implemented. We explored the locations to be suggested, collected their pictures and stored them in the static folder of our project. After implementation of the front end for this feature, we connected it with the backend to complete it.

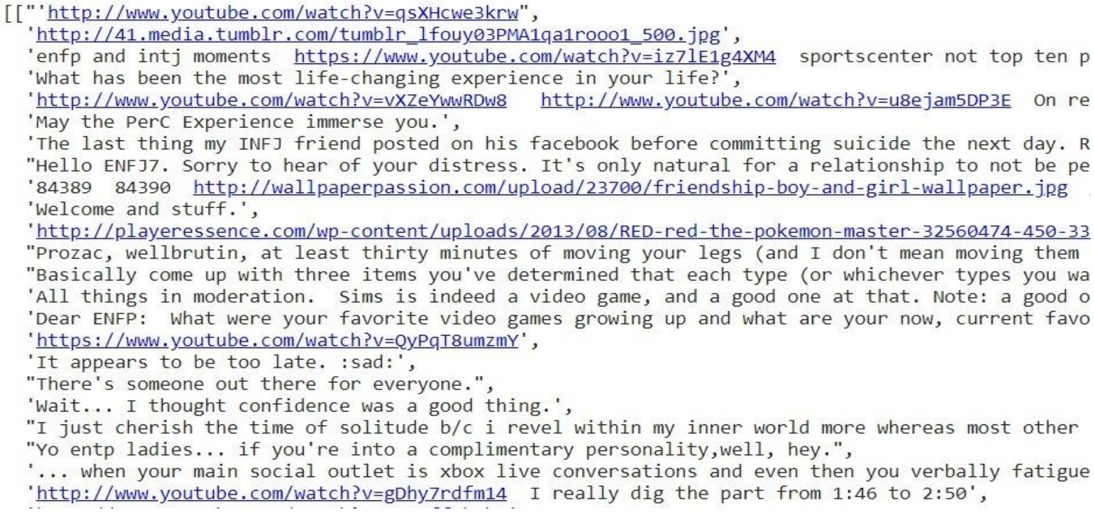
Then, we proceeded with the film viewer part of our project. We built the dashboard for the film viewers portal and then started with our feature implementation. We collected film details such as their title, overview and stored them in our database table. We also stored genre preferences of the users in our database. We started out with front end development first and then implemented the backend logic to give film recommendations based on user input. After connecting the two ends, we tested this feature with various test cases to make sure it is working as expected.

After this, we went back to the filmmaker side to implement the last feature of viewer preferences as this has dependency on the film viewers portal. We used simple backend logic to retrieve genre preferences of the users and displayed it graphically in the front end.

After completing implementation of all our project features, we then hardened our code to work for invalid inputs and other minor defects.

The dataset was firstly preprocessed to convert the independent attributes into numeric values as required by the Machine learning model. We mapped the actors, actresses, director to our defined scores. For genres, we used One-hot encoding to get a 1dimensional vector representation of the genre. The Pan Indian release values Yes and No were mapped to 1 and 0 respectively. We stored the trained model in a file which is further used by the backend for making predictions for newer inputs.

Pre-Processing:



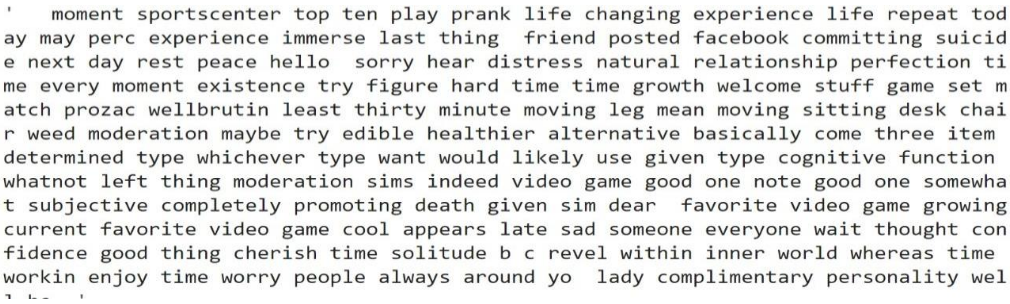
```
[["'http://www.youtube.com/watch?v=qsXHcwe3krw",
  'http://41.media.tumblr.com/tumblr_lfouy03PMA1qa1rooo1_500.jpg',
  'enfp and intj moments  https://www.youtube.com/watch?v=iz7lE1g4XM4  sportscenter not top ten p
  'What has been the most life-changing experience in your life?',
  'http://www.youtube.com/watch?v=vXZeYwwRDw8   http://www.youtube.com/watch?v=u8ejam5DP3E  On re
  'May the PerC Experience immerse you.',
  'The last thing my INFJ friend posted on his facebook before committing suicide the next day. R
  "Hello ENFJ7. Sorry to hear of your distress. It's only natural for a relationship to not be pe
  '84389  84390  http://wallpaperpassion.com/upload/23700/friendship-boy-and-girl-wallpaper.jpg
  'Welcome and stuff.',
  'http://playeressence.com/wp-content/uploads/2013/08/RED-red-the-pokemon-master-32560474-450-33
  "Prozac, wellbrutin, at least thirty minutes of moving your legs (and I don't mean moving them
  "Basically come up with three items you've determined that each type (or whichever types you wa
  'All things in moderation.  Sims is indeed a video game, and a good one at that. Note: a good o
  'Dear ENFP:  What were your favorite video games growing up and what are your now, current favo
  'https://www.youtube.com/watch?v=QyPqT8umzmY',
  'It appears to be too late. :sad:',
  "There's someone out there for everyone.",
  'Wait... I thought confidence was a good thing.',
  "I just cherish the time of solitude b/c i revel within my inner world more whereas most other
  "Yo entp ladies... if you're into a complimentary personality,well, hey.",
  '... when your main social outlet is xbox live conversations and even then you verbally fatigue
  'http://www.youtube.com/watch?v=gDhy7rdfm14  I really dig the part from 1:46 to 2:50',
```

Fig 2. Raw dataset

The text from the dataset need to be processed before using and it is done in following steps

- Stop words present in the English language are stored in a list.
- Iterate through the entire dataset using the for loop
- From each entry the text is extracted and it is converted into lower case.
- Use the regular expressions to remove numbers, symbols and hyperlinks
- Use the Lemmatizer to create the lexemes from the words
- Check if the sentence contains stop words remove the stop words

The result of the above process is cleaned text.

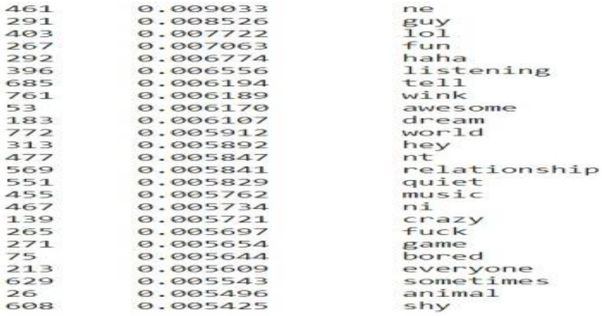


```
'   moment sportscenter top ten play prank life changing experience life repeat tod
ay may perc experience immerse last thing  friend posted facebook committing suicid
e next day rest peace hello  sorry hear distress natural relationship perfection ti
me every moment existence try figure hard time time growth welcome stuff game set m
atch prozac wellbrutin least thirty minute moving leg mean moving sitting desk chai
r weed moderation maybe try edible healthier alternative basically come three item
determined type whichever type want would likely use given type cognitive function
whatnot left thing moderation sims indeed video game good one note good one somewha
t subjective completely promoting death given sim dear  favorite video game growing
current favorite video game cool appears late sad someone everyone wait thought con
fidence good thing cherish time solitude b c revel within inner world whereas time
workin enjoy time worry people always around yo  lady complimentary personality wel
```

Fig 3. Cleaned Text after Preprocessing

Vectorization/Encoding:

The text data(corpus) consists of so many words and should be summarized to a few keywords only. In the end, we want some method to compute the importance of each word.One way to approach this would be to count the no. of times a word appears in a document. So, the word importance is directly proportional to its frequency. This method is, therefore, called Term Frequency(TF).Few words can appear so many times in a particular document so the important feature cannot be directly selected based on the no.of times it appears in the document, the uniqueness of word should also be taken into consideration and it is done using the Inverse Document Frequency(IDF) method.



```
461       0.009033          ne
291       0.008526          guy
403       0.007722          lol
267       0.007063          fun
292       0.006774          haha
396       0.006556          listening
685       0.006194          tell
761       0.006189          wink
53        0.006170          awesome
183       0.006107          dream
772       0.005912          world
313       0.005892          hey
477       0.005847          nt
569       0.005841          relationship
551       0.005829          quiet
455       0.005762          music
467       0.005734          ni
139       0.005721          crazy
265       0.005697          fuck
271       0.005654          game
75        0.005644          bored
213       0.005609          everyone
629       0.005543          sometimes
26        0.005496          animal
608       0.005425          shy
```

Fig 4. Top 25 important Features

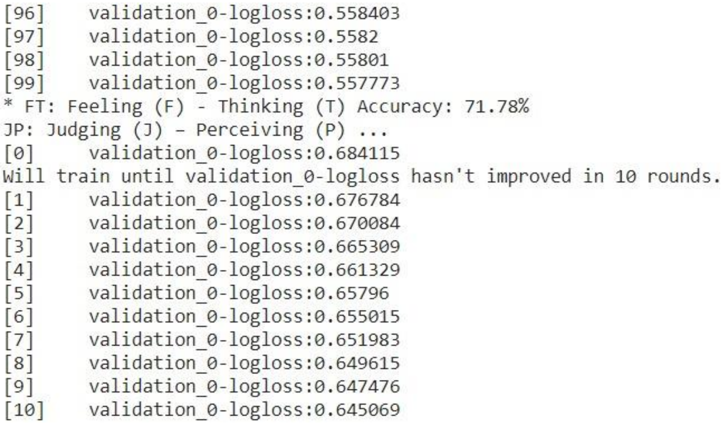
Figure 4 shows the top 25 important features from the total list of features.
The text data is converted into the numerical in the following steps:

- Cleaned text to a matrix of token counts using the countvectorizer
- Learn the vocabulary dictionary and return term-document matrix
- Transform the count matrix to a normalized tf or tf-idf representation using TFIDF Transformer
- Learn the idf vector (fit) and transform a count matrix to a tf-idf representation

The result after the above process is the encoded feature matrix .

Training the XGBoost model:

- Early Stopping and Performance Monitoring is done while training XGBoost with each classifier model to avoid overfitting.
- The validation Metric used here is Logarithmic Loss.
- As shown in Figure 5, Early Stopping is done for every 10 rounds if the performance hasn't improved in 10 rounds or else the performance is monitored upto 100 rounds and the best value of the metric is taken and the corresponding accuracies are obtained for each classifier.



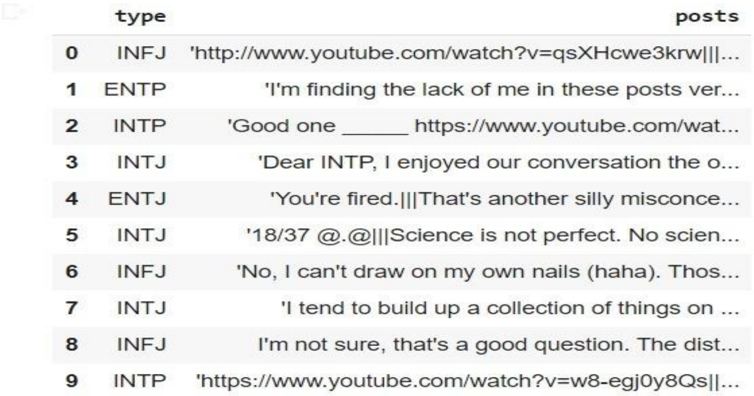
```
[96]    validation_0-logloss:0.558403
[97]    validation_0-logloss:0.5582
[98]    validation_0-logloss:0.55801
[99]    validation_0-logloss:0.557773
* FT: Feeling (F) - Thinking (T) Accuracy: 71.78%
JP: Judging (J) – Perceiving (P) ...
[0]     validation_0-logloss:0.684115
Will train until validation_0-logloss hasn't improved in 10 rounds.
[1]     validation_0-logloss:0.676784
[2]     validation_0-logloss:0.670084
[3]     validation_0-logloss:0.665309
[4]     validation_0-logloss:0.661329
[5]     validation_0-logloss:0.65796
[6]     validation_0-logloss:0.655015
[7]     validation_0-logloss:0.651983
[8]     validation_0-logloss:0.649615
[9]     validation_0-logloss:0.647476
[10]    validation_0-logloss:0.645069
```

Fig 5. Early Stopping and Performance Monitoring

- To increase the model's performance further, Hyperparameter Tuning is performed for XGBoost and the best parameter values are obtained.
- These values are then used to train the model in order to achieve the best possible accuracy.
- Hyperparameter tuning is performed by using Grid Search CV method here by using K Fold cross Validation where the dataset is split into k parts, k is 10 for our model training
- The accuracy metrics are used to check the performance of the model along with the roc curve.
- To compare the XGBoost model, Logistics Regression ,Naïve Bayes, Decision Tree Classifier are also trained with the same dataset.

Dataset Description

The data set used in the paper is the mbti dataset which is publicly available in the Kaggle. The dataset contains 8675 rows/entries. There are two columns the first being the personality type which is a word of four letters where each letter represents a personality type and the second column is the combination of 50 social media posts corresponding to that particular user separated with the pipeline character. This information contained in the dataset cannot be directly used as it contains http links, numbers, symbols and stop words which has no impact on the personality of the user. The representation of the personality as said above is not direct; all the traits are combined to generate a single personality type. For better processing of the entries in the binary classification needed one character at a time and it should be as per what model is running and what are the expected traits . In order to make the above process easier four extra columns are added to the dataset where each column is a binary classification result. The dataset obtained is not so balanced either. There is high class imbalance in the data set there are so many entries of the Extrovert compared to the Introvert, number of entries of sensing are significantly more than the number of intuition entries, similarly the number of entries of Judging are more than the number of entries of the Perceiving.



	type	posts
0	INFJ	'http://www.youtube.com/watch?v=qsXHcwe3krw|||...
1	ENTP	'I'm finding the lack of me in these posts ver...
2	INTP	'Good one ______ https://www.youtube.com/wat...
3	INTJ	'Dear INTP, I enjoyed our conversation the o...
4	ENTJ	'You're fired.|||That's another silly misconce...
5	INTJ	'18/37 @.@|||Science is not perfect. No scien...
6	INFJ	'No, I can't draw on my own nails (haha). Thos...
7	INTJ	'I tend to build up a collection of things on ...
8	INFJ	I'm not sure, that's a good question. The dist...
9	INTP	'https://www.youtube.com/watch?v=w8-egj0y8Qs||...

Fig 6. Dataset Entries

5 RESULTS

Results of each classifier are displayed and accuracies of each classifier are compared with other models

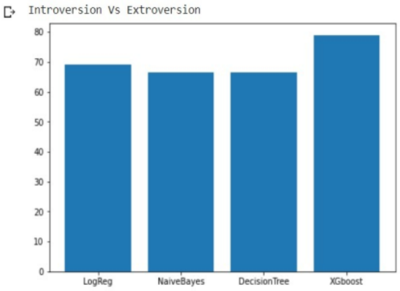


Fig 7. Introversion vs Extroversion Performance Bar Graph

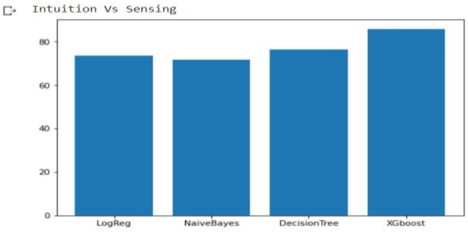


Fig 8. Intuition vs Sensing Performance Bar Graph

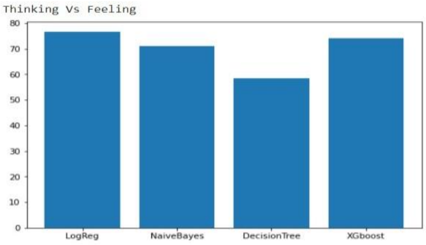


Fig 9. Thinking vs Feeling Performance Bar Graph

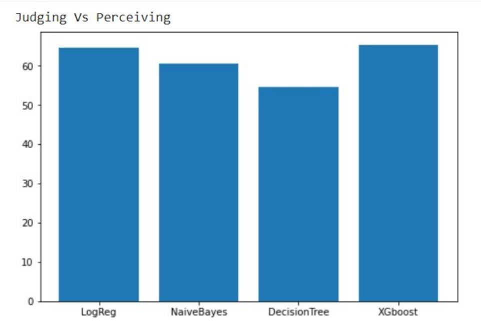
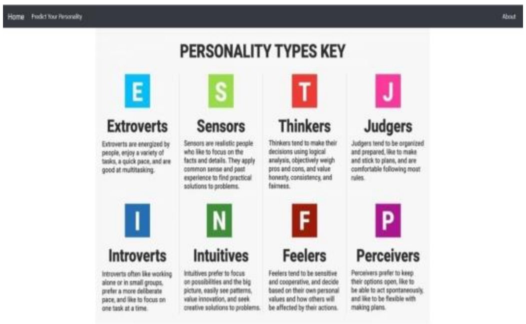


Fig 10. Judging vs Perceiving Performance Bar Graph

Above Figures show the accuracies received and observed and bar graphs are plotted for the four binary classifiers. For Introversion Vs Extroversion, the highest accuracy was obtained from XGBoost followed by Logistic Regression, Naïve Bayes and Decision Tree. For Intuition vs Sensing, highest accuracy was obtained from XGBoost followed by Decision Tree, Logistic Regression and Naïve bayes. For Thinking vs Feeling, Logistic regression gave the highest accuracy followed by XGBoost, Naïve Bayes and Decision Tree.



PERSONALITY TYPES KEY

Extroverts

Extroverts are energized by people, enjoy a variety of tasks, a quick pace, and are good at multitasking.

Sensors

Sensors are realistic people who like to focus on the facts and details. They apply common sense and past experience to find practical solutions to problems.

Thinkers

Thinkers tend to make their decisions using logical analysis, objectively weigh pros and cons, and value honesty, consistency, and fairness.

Judgers

Judgers tend to be organized and prepared, like to make and stick to plans, and are comfortable following most rules.

Introverts

Introverts often like working alone or in small groups, prefer a more deliberate pace, and like to focus on one task at a time.

Intuitives

Intuitives prefer to focus on possibilities and the big picture, easily see patterns, value innovation, and seek creative solutions to problems.

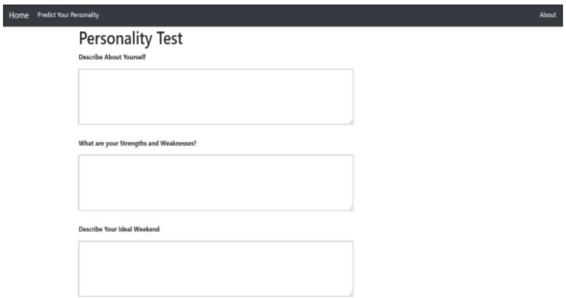
Feelers

Feelers tend to be sensitive and cooperative, and decide based on their own personal values and how others will be affected by their actions.

Perceivers

Perceivers prefer to keep their options open, like to be able to act spontaneously, and like to be flexible with making plans.

Fig 11: Home page of the website that give brief description about the mbti personality characteristics



Personality Test

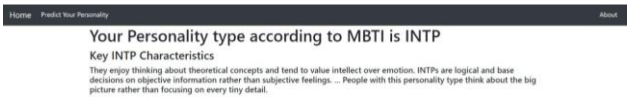
Describe About Yourself

What are your Strengths and Weaknesses?

Describe Your Ideal Weekend

Fig 12: Question Form

The questions form where the user answers the questions. The text entered in all the text area fields are combined to form a single text input against which the model is run.



Your Personality type according to MBTI is INTP

Key INTP Characteristics

They enjoy thinking about theoretical concepts and tend to value intellect over emotion. INTPs are logical and base decisions on objective information rather than subjective feelings. ... People with this personality type think about the big picture rather than focusing on every tiny detail.

Fig 13 is the result page where the personality type of the user along with what the user might like to do and the characteristics of the user are printed.

6. CONCLUSIONS AND FUTURE SCOPE

The processing of the data is completed using the nltk library which provides the inbuilt list of stop words available in the English language. The regular expressions are used to remove the http links, symbols, numbers. Using the TF-IDF vectorizer feature vectors are created with the low computational overhead. The proposed system for the personality prediction using the XGBoost yielded the best performance. The XGBoost model outperformed the logistic regression model, Naïve bayes Classifier and Decision Tree Classifier. The performance of the XGBoost is increased after the hyper parameter tuning which further increased the accuracy difference between the XGBoost and the other models. The highest of the XGBoost model is 86 for the classifier Intuition vs Sensing.The dataset is highly imbalanced that limits the performance of the system. The system predicts the results purely based on the answers entered by the user at the current time so the user's presence of mind while answering the questions plays a greater impact on the result.

In future the proposed system can be implemented in the real time applications like career advices where one can take advice like what are the best career choices for a person with a particular personality traits for example an extrovert can be good at jobs involving lot of communication whereas the introverted people might be comfortable with jobs involving minimal communication , movie / music recommendations where a particular personality type people can like a certain genre of movies or music, blog websites where the users can meet the similar minds.

REFERENCES

[1] P. Vimala Manohara Ruth, Dr. Y. Rama Devi, E. Haritha, N. Shiva Kumar, “Prediction of phishing website for data security using various machine learning algorithms”, International Journal of Creative Research Thoughts, Vol 9, issue 6, June 2021.

[2] A. V. Kunte and S. Panicker, "Using textual data for Personality Prediction:A Machine Learning Approach," 2019 4th International Conference on Information Systems and Computer Networks (ISCON), 2019, pp. 529-533, doi: 10.1109/ISCON47742.2019.9036220.

[3] Brandon Cui, Calvin Qi, “Survey Analysis of Machine Learning Methods for Natural Language Processing for MBTI Personality Type Prediction”, Stanford,2018

[4] Hernandez, Rayne, Knight, Ian Scott, “Predicting Myers-Briggs Type Indicator with text classification”,31st Conference on Neural Information Processing System, USA,2017

[5] S. Bharadwaj, S. Sridhar, R. Choudhary and R. Srinath, "Persona Traits Identification based on Myers-Briggs Type Indicator(MBTI) - A Text Classification Approach," 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2018, pp. 1076-1082, doi: 10.1109/ICACCI.2018.8554828.

[6] J. Golbeck, C. Robles, M. Edmondson and K. Turner, "Predicting Personality from Twitter," 2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing, 2011, pp. 149-156, doi: 10.1109/PASSAT/SocialCom.2011.33.

[7] B. Y. Pratama and R. Sarno, "Personality classification based on Twitter text using Naive Bayes, KNN and SVM," 2015 International Conference on Data and Software Engineering (ICoDSE), 2015, pp. 170-174, doi: 10.1109/ICODSE.2015.7436992.

[8] Tommy Tandera, Hendro Derwin Suhartono, Rini Wongso and Yen Lina Prasetio, "Personality Prediction System from Facebook Users", 2nd International Conference on Computer Science and Computational Intelligence, 13–14 October 2017.

[9] Agrawal, K., Bhargav, G., Spandana, E. (2021). Diabetes Diagnosis Prediction Using Ensemble Approach. In: Nath, V., Mandal, J.K. (eds) Proceedings of the Fourth International Conference on Microelectronics, Computing and Communication Systems. Lecture Notes in Electrical Engineering, vol 673. Springer, Singapore. https://doi.org/10.1007/978-981-15-5546-6_66

[10] T. Prathima, B Anjana, V Apoorva, BR SreedharEnsemble Based Hybrid Recommender Systems International Journal of Innovative Technology and Exploring Engineering (IJITEE) 826-833 Jan-2020 10.35940/ijitee.C8460.019320

ACKNOWLEDGEMENT

The authors would like to acknowledge Sai Raj Yadav Saanam and Supreet V for their contribution and support.