American Journal of Sciences and Engineering Research

e-ISSN -2348 – 703X, Volume 4, Issue 4, 2021

Enhanced Movie Prediction Model Using Data Mining PHP

Eze, Chinyere[1], Onuodu, Friday Eleonu[2]

Research Scholar, Department of Computer Science, Ignatius Ajuru University of Education, Rivers State, Nigeria.
Visiting Scholar, Department of Computer Science, University of Port-Harcourt, Rivers State Nigeria.

ABSTRACT: *The movie industry is one of the fastest growing industry across the world. This industry, both on the local and international scale has gained tremendous attention of investors and analysts due to the increasing interest and acceptance which increasingly greets the industry. There is however a serious concern about the outcome of the contents which are released from the industry as it affects revenue, profit and recognition both for the producers, promoters, investors and viewers. Researchers have tried to solve this problem by developing systems and models using various machine learning algorithms to forecast the success of movies before they are released however, their researches have met with constraints such as ambiguity, poor datasets, poor feature selection and time complexities. In this work, we have developed an enhanced movie prediction model using data mining PHP. We used two datasets from IMDb and Box office Mojo for our analysis and prediction. The dataset was split into 80% training set and 20% of test set. The Gaussian Classifier was trained and tested using the cleaned dataset and Artificial Neural network (ANN). The Spiral methodology was adopted in this approach. The model was implemented using Hypertext Preprocessor for data collection and Python 3.7 for data analysis and prediction. Our model was evaluated against the existing model and our model had an overall accuracy of 97% while the existing model had 77%. This study could be beneficial to movie investors, to movie directors, movie producers, to movie promoters, to movie viewers and to the research community.*

Keywords: *Data Mining, Movie Prediction, PHP, Movie, Reviews, Box office, Sentiments, Machine Learning.*

I. Background to the Study

The movie industry is among the few industries that has greatly evolved with the growing technological advancements. Concepts such as trailers, live streaming, movie ratings and reviews are some of features that technologies has made available to make movies more interesting for the younger generation.

On the side of the movie industries, investors and directors, the interest is on the revenue that will be generated from a movie after it is released, and no necessarily on how interesting the movie is. These stakeholders always seek for a way to predict the level of success a movie will have when released which is directly proportional to the revenue, as this information will help them make vital information about the production costs of the movie and box office price tag on the movie.

The success of a movie simply means the level of satisfaction of an audience expressed in reviews about a movie, actors, scenes, plot etc. Factors such as the actors, genre, director, writer, plot etc. are directly responsible for the success of a movie. Other factors include IMDb rating, IMDb metascore, IMDb vote count, rotten tomatoes' tomatometer, actors and director social fan following, Wikipedia views and trailer views [1]. This means that if a

Received- 21-07-2021, Accepted- 03-08-2021

particular actor(s) who featured in previous movies which were amazing features in a new movie which is about to be released, the movie will be highly anticipated for and the success rate will very much likely be high, because the audience will give positive reviews even before viewing that movie. This is the same with the other factors of movie success prediction too.

Researcher have developed several prediction models for movie success using several machine learning algorithms. Data mining in PHP is the algorithm that is proposed in this study to improve on movie prediction using datasets from IMDb repository.

Data mining approach will be used to extract he relevant movie data from the large data pool for analysis and prediction or classification of the success of future movies which have either been half produces and have their data on the net or movies which are about to be produced.

1.1. Statement of the Problem

Several researches have been carried out in the field of movie success prediction over the years. However, a close review of these systems exposed some limitations such as poor choice of dataset. This factors greatly affects the success of a movie as key factors that will be used for the prediction will be missing. Sometimes the dataset is also small. Another identified limitation is lack of efficient prediction parameters/features for movie success. These are the issues we hope to address in our research.

1.2. Aim and Objectives

The aim of this work is to develop an enhanced movie success prediction model using data mining PHP. Our specific objectives are to:

i. design a prediction model for movie success using movie historical dataset from IMDb.

ii. train our model using Artificial Neural Network (ANN). Training set is 70%.

iii. test our prediction model using 30% of the dataset.

iv. implement using Hypertext preprocessor for Data mining and Python for Data Analysis and Prediction.

II. Literature Review

This section provides a survey of the basic concepts that form our research.

1.3. Movie Success Prediction: An Overview

Movie success is the level of satisfaction, revenue and reviews which a movie generates after its release. The box office is a place where movie tickets are bought when a movie is to be shown in a cinema. Therefore, this makes the box office a direct source of generating information about movie success. Movie success prediction is a major focus of the entertainment industry especially the stakeholders of the movie industries such as investors. The success rate of a movie will determine the amount of revenue that will be generated from the movie which will in turn determine the profit on the side of the investors. This is a key detail that will enforce informed decision making before a movie will be produced.

I. Factors of Movie Success and Their Roles

The factors responsible for movie success are divided into 2 broad groups namely Classical Movie Attributes and User anticipation and Response.

a. Classical Movie Attributes

This group is made up of the cast, director, producer, and genre of a movie and they play a crucial role in the movie's success. Most times before an audience makes the decision to see a movie, the research on the actors, director, producer, genre, production studio of the movie and after getting this data, they decide on the best movie to see. This is due to the fact that, generally, it is believed that these factors are what will guarantee the awesomeness of the movie. For example, Netflix movies are highly rated as awesome movies and once a viewer sees that a movie is from Netflix, there will be increased interest in the movie.

b. User Anticipation and Response

This includes the reviews and response of users on social media. This feedback increases the success rate of movies and helps to predict movie success. The social media is a world were several views and opinions are aired daily about people, products and services. It is the largest source of opinion pool as millions are generated every day from people all over the world. Movie reviews are not left on these platforms. It is from these social media platforms that people decide on the next movie to see based on reviews from IMDb, users, YouTube etc. Usually, people who have seen these movies comment on them and thereby generate a lot of useful information for informed decision on the awesomeness or not of the movie.

1.4. Sources of Movie Review Data

Movie review data can be gotten from several platforms ranging from the social networks to other online huge movie data stores. Data can be gotten from these sources using their APIs or other web crawling tolls in order to generate the desired data.

I. Movie Databases

Sources like TMDb and OMDb expose their APIs to retrieve IMDB movie. To start with these APIs, IMDb ids of the movies have to be collected. For example, to get the list of the ids of the movies released in late 2018 or later, an IMDb list of movies can scraped using a script based on their ids. These ids can be used to get movie related data such as directors, writers, cast, runtime, votes, ratings, etc.

II. Wikipedia Data

This platform provides data on the popularity of movies and this data can be collected through Wikimedia REST API for Wikipedia page views of the movie 30 before the release date of the movie. For e.g. to retrieve views on a movie called Game of Thrones from the API, Wikipedia links of the form "Game of Thrones (2019_film)" is required for the movie. So these links is first scraped using Google search and then used with the API.

III. YouTube Data

Here, data on the buzz created from a movie is gotten from YouTube's API. This data can be used to analyze likes, shares, and comments of the official trailer and teaser. Again, the API will require the video id to get any sort of information on the videos. The YouTube video ids of the trailers and teasers of the movies, which are uploaded before the movie is released, will be scraped if the data is to be used for future prediction using Python scripts. And then the comments and other video statistics are collected through API.

IV. Social Media

Social media platforms such as Facebook, Twitter, Instagram, WhatsApp etc. are huge sources of movie review data especially, Facebook and Twitter. These platforms draw a lot of users talking about different subjects all over the world every second of the day. Twitter API provides the required movie review data and is scrapped using "Tweepy" a data scraping tool. Facebook API is also used to get movie review and rating data using web scraping tools as well.

1.5. Data Mining

There is a large pool of data for every data type of data which a data analyst may be interested in. However, not all the data are relevant to the issue of discuss. Therefore, there must be a way of collecting only the necessary and related data for a particular analysis, this is where data mining comes in. Data mining is a process of extracting useful data from a larger set of raw data.

I. Techniques of Data Mining

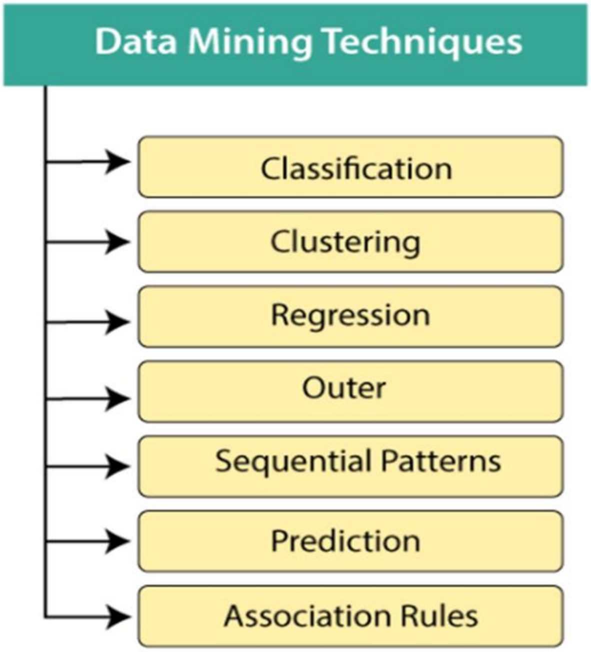


Figure 1: Techniques of Data Mining (Source: Tutorial Point)

a. **Classification:** This technique is concerned with the analysis of the attributes associated with different types of data. These attributes are then categorized into different groups and this groups will help the analyst understand the data better.

b. **Clustering:** This technique is very similar to the classification except for the fact that in clustering attributes with similar qualities are grouped together in large chunk.

c. **Regression:** This technique is used primarily as a form of planning and modeling, is used to identify the likelihood of a certain variable, given the presence of other variables. More specifically, regression's main focus is to uncover the exact relationship between two (or more) variables in a given data set.

d. **Outlier Detection:** This technique is used to simply recognize the overarching pattern and provide a clear understanding of your data set. Anomalies are also identified using his technique or outliers in your data.

e. **Sequential Pattern:** One of the most basic techniques in data mining is learning to recognize patterns in your data sets. This is usually a recognition of some aberration in your data happening at regular intervals, or an ebb and flow of a certain variable over time.

f. **Prediction:** Prediction is one of the most valuable data mining techniques, since it's used to project the types of data that will be seen in the future. In many cases, just recognizing and understanding historical trends is enough to chart a somewhat accurate prediction of what will happen in the future.

g. **Association Rules:** Association is related to tracking patterns, but is more specific to dependently linked variables. In this case, specific events or attributes that are highly correlated with another event or attribute are searched for.

1.6. Related Work

Chakraborty et al [1] proposed movie success prediction using historical and current data mining. The model was developed to predict the success of upcoming movies using several factors. Each factor was assigned by a weight and success/failure of the upcoming movies was predicted based on the factor's value. However, they could not implement using data mining PHP.

Sanjai et al [2] proposed prediction of movie success for real world movie success. They built a prediction engine based on classification and fuzzy logic to categorize a movie as either successful or not. They also developed an algorithm to calculate the IMDb score of a movie based on certain parameters. They used 7 different categories to classify the movies as: flop, Bad, Watchable, Decent, Good, Great and Amazing. However, they did not performance an evaluation of their system to demonstrate its prediction accuracy.

Bhave et al [3] proposed role of different factors in predicting movie success. Some of the factors analyzed include classical movie attributes and user anticipation and response. They suggested that if more data is taken into account and properly integrated, then greater accuracy could be achieved than considering the classical or social factors individually. However, they could not integrate the classical and social factors and using the basis of interrelation among the classical factors to assign the weights which will provide higher accuracy.

Apala et al [4] proposed prediction of movies box office performance using social media. They used data mining tools to generate patterns for predicting box office performance of movies via data collected from several social media and web sources such as Twitter, YouTube and the IMDb. The prediction was centered on decision factors gotten from a historical movie database, followers count from Twitter, and sentiment analysis of YouTube viewers' comments. The prediction was labeled in three classes, Hit, Neutral and Flop, using Weka's K-Means clustering tool. Interesting patterns for prediction were generated by Weka's J48. However, they did not carry out performance evaluation to show the efficiency of their model.

Mhowwala et al [5] proposed movie rating prediction using ensemble learning algorithms. They collected data from different sources such as YouTube and IMDb and used two ensemble classifiers (Random Forest and XGBoost) on the dataset to predict a continuous output. The result showed that XGBoost being a more efficient algorithm performed better when compared to random forest. However, the size of dataset was too small to attain the required accuracy.

Suthana and Ramasamy [6] proposed context-based classification reviews using association rule mining, fuzzy logics and ontology. They used text mining techniques to tag the given context and to offer the connectivity between the words in the review context and to provide the semantic similarity between the contexts. Textual analysis was performed using combination of association rules and ontology mining. They compared relation between review and their context using the semantic analyzer based on the fuzzy rules. However, they could not implement the developed model in a visualization model.

Rahim et al [7] proposed mining trailers data from YouTube for predicting gross income of movies. They sourced dataset of 7988 movie trailers from YouTube. Attributes such as number of views, number of likes, number of dislikes, and number of comments were in the dataset. Two prediction models were developed and four regression techniques were applied to find out the most suitable technique for predicting the gross income of a movie. From the comparative analysis, linear regression was depicted to be the most suitable method. However, they model is unrobust.

Wu et al [8] proposed movie box office prediction based on ensemble learning. They collected film data from 1980 to 2018 from box office mojo. Using machine learning and ensemble learning, they built a predictive model. From the experimental result, gradient boosting decision tree (GBDT) gave the best performance of R_2 higher than 0.995. However, they could not predict the success of an upcoming movie using their model.

Chauhan et al [9] presented movie success prediction using data mining. They used the R software to predict success or failure of a movie based on several attributes, and criteria on which the success of any movie could depend. The proposed work aimed to develop a system based on data mining methods that would aid in predicting the success or failure of a movie thereby reducing certain level of uncertainty of future of a movie. However, they could not use more attributes to build their predictive model in order to increase efficiency.

Meenaskshi et al [10] proposed a data mining technique for analyzing and predicting the success of movie. They aimed at predicting the performance of past as well as the future of movie for the purpose of business certainty or simply a theoretical condition in which decision making can be enhanced. From the Experiment results, they observed that applying data mining techniques to the data in the IMDb dataset was difficult and that data processing was difficult. However, the result was not generalizable and could not be used for informed decision making.

Mahmud et al [11] proposed a machine learning approach to predict movie revenue based on pre-released movie metadata. Different machine learning algorithms like Logistic Regression, Support Vector Machine (SVM) and Multi-Layer Perceptron (MLP) were used to predict the box office return of a movie based on the data available before the release of the movie. The models used 35 movie parameters from 3200 movies as inputs to predict the profit made by a movie and classify the success of a movie from “flop” to “blockbuster” based on the generated revenue. Accuracy of 85.31% was gotten from their performance evaluation.

Patil et al [12] proposed box office movie prediction system. The system predicts an approximate success rate of a movie based on its profitability by analyzing historical data from different sources like International Movie Database, Rotten Tomatoes, Box Office Mojo and Metacritic. Using Support Vector Machine (SVM), Neural Network and Natural Language Processing the system predicts a movie box office will be a hit or flop based on some pre-released features and post-released features. However, their model cannot be applied for upcoming movies.

Upadhyay et al [13] proposed movie success prediction using data mining. They presented the use of a review system in predicting the success rate of a movie. They employed sentiment analysis to obtain Moviegoers’ opinions of a movie before and after the release of the movie. A custom dictionary was developed comprising words commonly used in movie reviews, which was mapped to their corresponding weight-age in order to score reviews on a scale of one to five, and accordingly classify the success rate of movies. However, they could not develop a robust model that will accommodate reviews from other sources.

Brar and Sharma [14] presented sentiment analysis of movie review using supervised machine learning techniques. Hey carried out sentiment analysis using feature based opinion mining and supervised machine learning. Noun, verbs and adjectives were used as opinion words to determine the polarity of reviews and classify them hem as negative or positive. Natural Language Processing Toolkit was used to tag parts of speech. However, they could not be used to mine reviews from smart phone and other devices.

Caghyor et al [15] presented forecasting box office performance using machine learning algorithms. They designed a forecast model using different machine learning algorithms such as SVM, Decision Tree Regression, ANN and Linear Regression to estimate the theatrical success of US movies in Turkey before their market entry and they evaluated the four models. They investigated MPAA rating, budget, star and director power, sequel, adaptation, number of screens, domestic performance, release time lag between domestic and foreign market as independent variables. However, their model could not classify the movies based on their success rates.

Kim et al [16] proposed finding Nemo. Nemo is a machine learning method for predicting movie performances. They used the selective and important predictive variables for ROI by using the Bayesian variable selection method. This removed measurement error in the Hollywood dataset, and unnecessary statistical conditions such as multi-collinearity and independence among the explanatory variables for ROI. Our results showed that the neural network Model for ROI is overall superior to the well-known machine learning methods in terms of RMSE. However, this model is ambiguous and cannot be used by simple organizations.

Ponce et al [17] presented a study on data mining in web applications. They explored the various applications of Data mining for web environment and applications. Then they proposed and implemented a biometric verification system based on data gotten form the cloud to ascertain the identity of a person who attempts to access a restricted part. However, they could not apply in their research in prediction of future events.

Ramageri [18] presented a study on data mining and techniques. They surveyed various algorithms such as SVM, Decision Tree, Bayesian Classification, Clustering, Genetic Algorithm, Nearest Neighbor etc. extensively and stated the benefits of using each of the algorithms. However, there was not implementation to demonstrate the behavior of each of these algorithms.

Quader et al [19] proposed a machine learning approach to predict movie box-office success. They presented a decision support system for movie investment sector. From their result, Neural Network gave the best prediction accuracy of 89.27% for upcoming movies. They also discovered that the factors with the highest determinant rate were IMDb votes and no. of screens.

Ahmad et al [20] presented a survey on machine learning techniques in movie revenue prediction. They discovered that regression, classification and clustering data mining approaches were used in the reviewed articles, with regression and classification carrying the largest share. Cast, number of screens, and genre, were identified as the most widely used features in movie revenue prediction. Multiple linear regression and support vector machines were the most efficient algorithms commonly used prediction algorithms, while mean absolute percentage error, root mean-square error, and average percentage hit rate were the evaluation metrics used the most. However, the result from this study cannot be generalizable.

III. Methodology

The Spiral methodology was adopted in this approach. Spiral methodology is a software development model introduced by Barry Boehm to decrease the uncertainty at each stage of software development. This model incorporates features of both the waterfall model and prototyping model. It is a risk-driven process model and its most important feature is to decrease the risk of the project.

IV. Analysis of the Existing System

The existing system was proposed by Meenakshi et al [10]. They proposed a data mining technique for analyzing and predicting the success of movie. They used the IMDb data which was extracted in a text format and converted to a CSV format. The dataset was made up of 600, 000 movie ratings which contained movie data such as film rank, number of votes and title of film. The data was finally stored in MySQL data base in form of tables. The data was cleaned by running SOL queries to remove redundant data and select attributes that will be necessary for prediction and analysis. They divided the dataset into training and test sets and applied K-means algorithm to develop models which was used for the test set. Decision trees were used for prediction of factors.

The system architecture of Meenakshi et al [10] is shown in figure 2.

Disadvantages of the Existing System

The authors did a very good job in predicting the movie success however, the system had the following drawbacks:

a. The result obtained from the research were not generalizable which implies that the system cannot be used in decision making.

b. The algorithms used in the training and prediction are unrobust and cannot handle different types of data efficiently.

c. The system lacks a good visualization model that will clearly demonstrate the prediction result in an understandable manner.
d. The system design is ambiguous and therefore suffers time complexity issues.

1.7. Algorithm of the Existing System

```
Step 1: Start
Step 2: Launch Java Movie Database
Step 3: Extract Movie.txt Files
Step 4: Convert .txt to Numeric files
Step 5: Store in Local Database (MySQL)
Step 6: Run SOL Query
        IF data = attributes THEN
                STORE data
                ELSE
        Drop row(no) WHERE data = data(value)
Step 7: Convert data to .CSV file.
Step 8: Divide data into Train & Test set
Step 9: Train data using k-means
Step 10: Test data using k-means model
Step 11: Perform Prediction using Decision trees
Step 12: End;
```

Analysis of the Proposed System

The proposed system is an enhancement of the existing system. In the proposed system two datasets were used. The datasets were gotten from Box Office Mojo and IMDb. IMDb dataset contains 10867 records with 20 attributes. The box office dataset was made up of 16543 records and 5 attributes including the title of movie, genre, director, budget, revenue, production studio, IMDb rating etc. The data was gotten by using PHP web crawling program to extract recent historical datasets with relevant attributes from the data stores.

Gaussian classifier was developed for the analysis of the data and Artificial neural network (ANN) was used to train and test the model. Training errors were minimal.

Data cleaning involved removal of repeated records, missing attributes and redundant attributes which do not directly influence the success of a movie and will not be directly used for prediction. Data splitting was carried out on the two datasets in a ratio of 80:20. 80% of the datasets were used for training while 20% was used for testing the model using “sklearn.model_selection” in Python. Also, as part of data cleaning, some data formats were converted into accurate formats to enable accurate analysis and prediction. E.g. the date format was correctly set for the “release-date” attribute.

For the classifier, we fed in features such as actor name, director name, IMDb rating, genre, profit, gross revenue and budget as the input. These attribute were carefully chosen because, they directly affect the success of a movie. The classifier was built to classify movies under 5 categories: Blockbuster (prediction =1), Interesting (prediction >=0.7), Average (prediction >=0.5), Watchable (prediction <=0.1), Horrible (prediction >=0.1). Table 1 clearly illustrates this prediction.

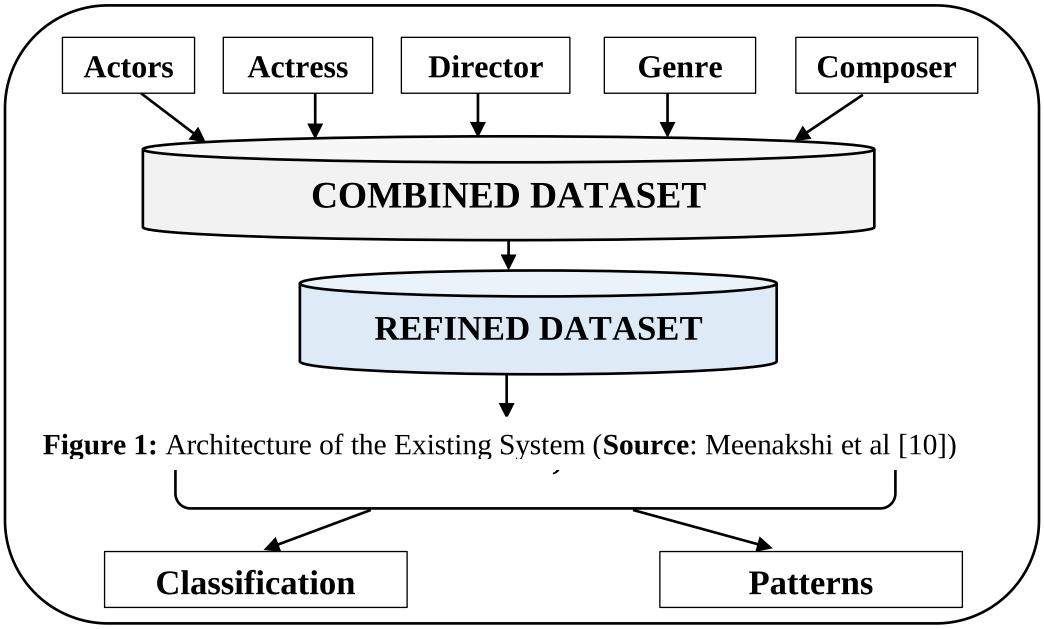
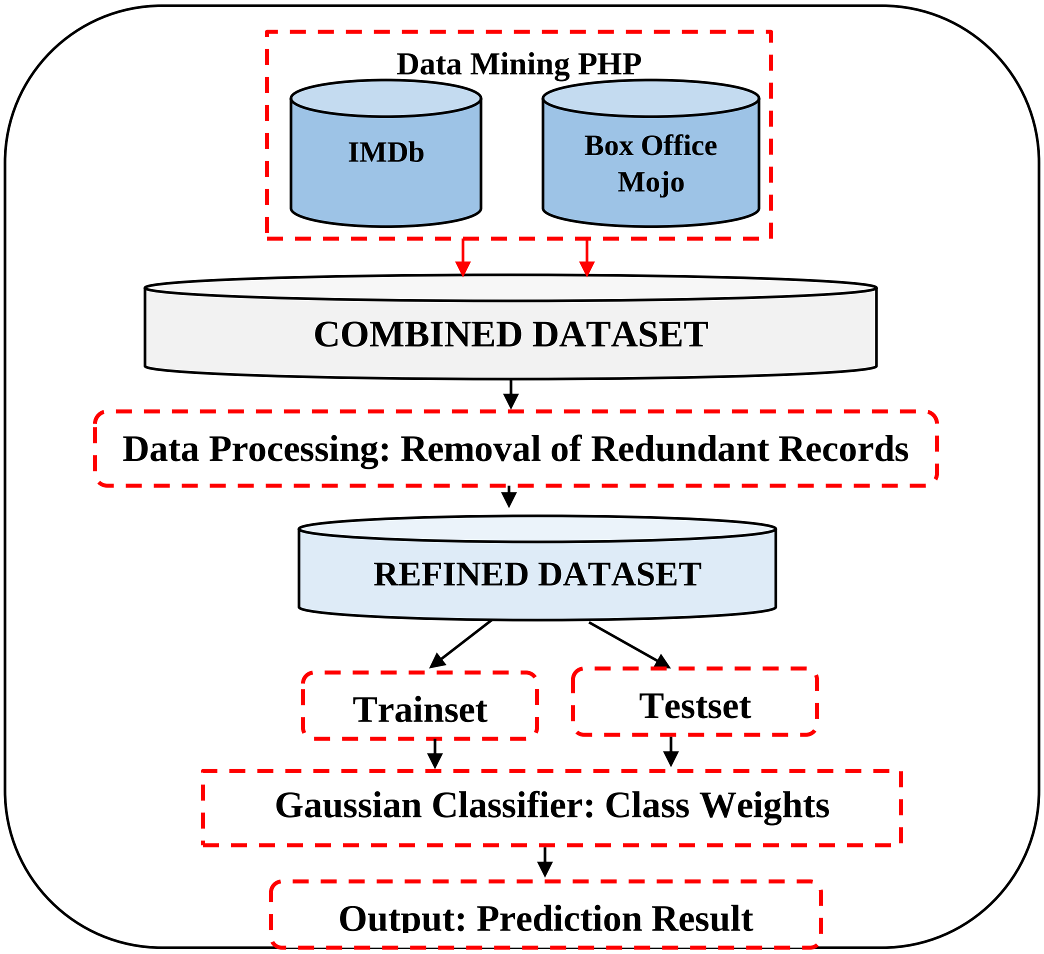


Figure 1: Architecture of the Existing System (**Source**: Meenakshi et al [10])



Data Mining PHP
IMDb
Box Office
Mojo
COMBINED DATASET
Data Processing: Removal of Redundant Records
REFINED DATASET
Trainset
Testset
Gaussian Classifier: Class Weights
Output: Prediction Result

Table 1: Class Distribution of Gaussian Classifier

S/N	Class	Class Score
1	Blockbuster	1
2	Interesting	0.7 – 0.99
3	Average	0.5 – 0.69
4	Watchable	0.2 – 0.45
5	Horrible	0.1 – 0.19

1.8. Advantages of the Proposed System

The merits of the proposed system over the existing include:

a. The result obtained from the research can be generalized for the prediction of any future movie which implies that the system can be used in decision making.

b. The algorithms used in the training and prediction are robust and can handle different types of data efficiently.

c. The system has a good visualization model that will clearly demonstrate the prediction result in an understandable manner.

d. The system design is simple and therefore does not have time complexity issues.

1.9. Algorithm of the Proposed System

```
Step 1: Start
Step 2: Launch PHP web Crawler
Step 3: Extract Movie.CSV Files
Step 4: Initiate Data Processing
    Data Processing = Redundancy Check + Empty Attribute Check
Step 5: Store in Local Database (MySQL)
Step 6: Run SOL Query
    IF data = attributes THEN
            STORE data
            ELSE
    Drop row(no) WHERE data = data(value)
Step 7: Launch Google Colab
Step 8: Import Python Libraries = Numpy +
        Pyplot + SkLearn + Pandas
Step 9: Import IMDb-movies.CSV +
        BoxOffice.CSV
Step 10: Split Data = Train set + Test Set
            Train set = 80%;
            Test set  = 20%
Step 11: Train Data using ANN
Step 12: Build Classifier = Gaussian Classifier
```

```
                  Train Classifier = Train set
                  Test Classifier      = Test Set
      Step 13: Set Class Weights
                  Blockbuster = 1;
                  Interesting   >=0.7
                  Average     >= 0.5
                  Watchable   >=0.2
                  Horrible     >= 0.1
Step 14: Perform Prediction
Step 15: Display Output;
Step 16: End:
```

V. Implementation

The system was using Python programming language on Google Colab real-time IDE. Python is a very important language when it comes to data science and prediction of future outcomes The Python libraries were imported to allow certain features or functions to be activated, such as pandas for storing the dataset.

1.10. Discussion of Results

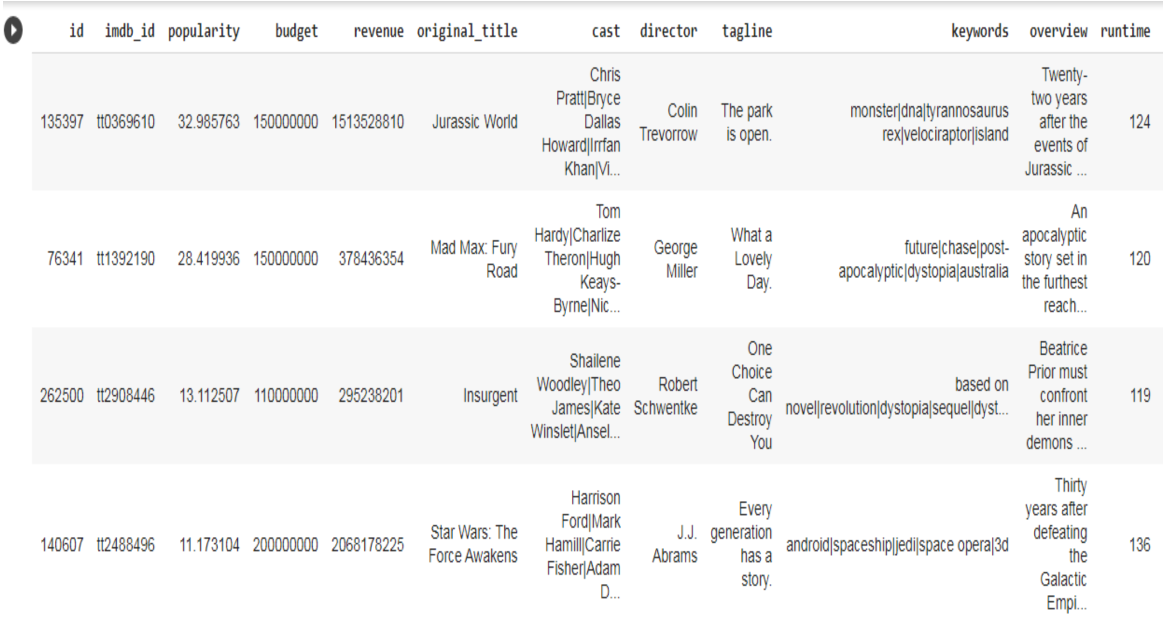
The data was imported and stored in variable name "df3" and "df4". Theses variable name were used to refer to the data throughout the implementation. The dataset was scanned to check for missing values, duplicate values and empty records. The search was completed and 1 row was identified as empty and remove, duplicate values were also removed and missing values were cleared. Some attributes were renamed and we changed their data type and format to suit the desired data type for the analysis. E.g. the release date was changed to "date" type. The first visualization of the cleaned dataset is shown in figure 4.

The data was split into 2 parts. The first part was renamed as train set which contained 80% of the data and the 2nd part was for the test data which contained 20% of the data.

The classifier was built using 7 features which grossly affect the success of a movie. They include director_name", "actor_name", "genres","imdb_score","budget","gross" and "profit_percent". The classifier was a Gaussian classifier represented as "GaussianNB" in Python. ANN was used to train the classifier using the train set and the test data was used to test the performance of the classifier.

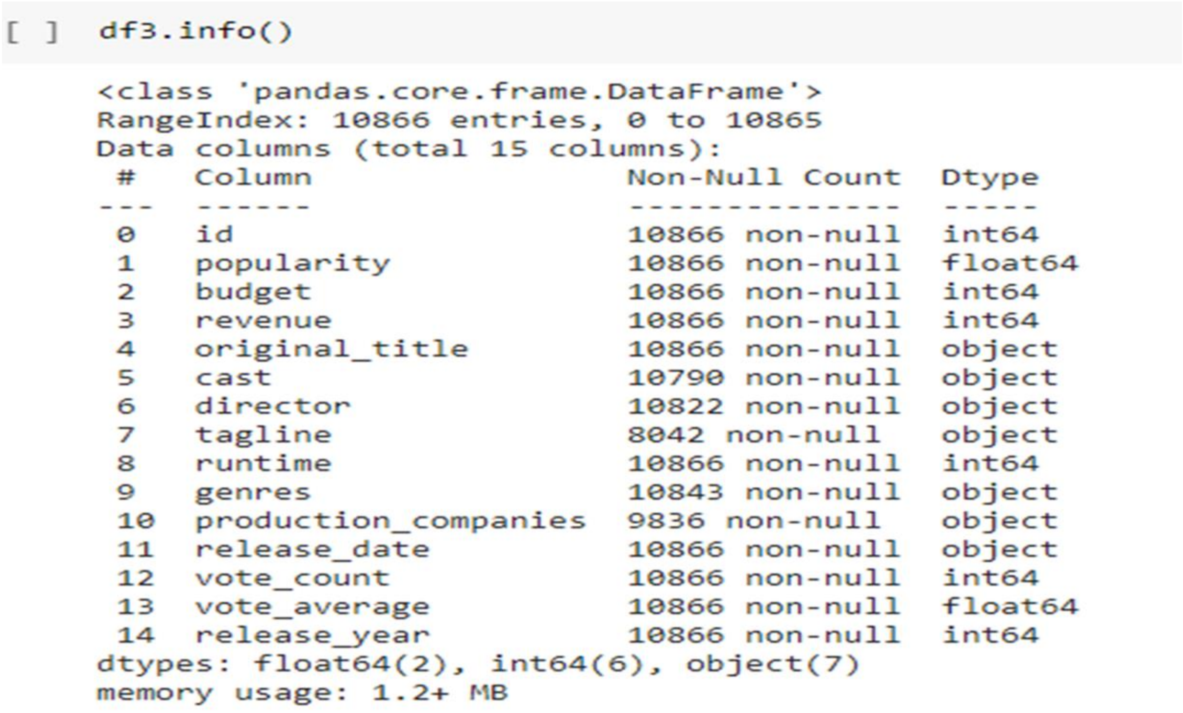
The features were fed as input into the classifier and 5 classes were to form the prediction output. The classes ranged from Blockbuster to horrible. Depending on how the movie will turn out to be. Weights were assigned to the classes to define the prediction boundaries that each class covers.

Fig. 4: Cleaned Dataset Free from Duplicates and Redundancy



id	imdb_id	popularity	budget	revenue	original_title	cast	director	tagline	keywords	overview	runtime
135397	tt0369610	32.985763	150000000	1513528810	Jurassic World	Chris Pratt|Bryce Dallas Howard|Irrfan Khan|Vi...	Colin Trevorrow	The park is open.	monster|dna|tyrannosaurus rex|velociraptor|island	Twenty-two years after the events of Jurassic ...	124
76341	tt1392190	28.419936	150000000	378436354	Mad Max: Fury Road	Tom Hardy|Charlize Theron|Hugh Keays-Byrne|Nic...	George Miller	What a Lovely Day.	future|chase|post-apocalyptic|dystopia|australia	An apocalyptic story set in the furthest reach...	120
262500	tt2908446	13.112507	110000000	295238201	Insurgent	Shailene Woodley|Theo James|Kate Winslet|Ansel...	Robert Schwentke	One Choice Can Destroy You	based on novel|revolution|dystopia|sequel|dyst...	Beatrice Prior must confront her inner demons ...	119
140607	tt2488496	11.173104	200000000	2068178225	Star Wars: The Force Awakens	Harrison Ford|Mark Hamill|Carrie Fisher|Adam D...	J.J. Abrams	Every generation has a story.	android|spaceship|jedi|space opera|3d	Thirty years after defeating the Galactic Empi...	136

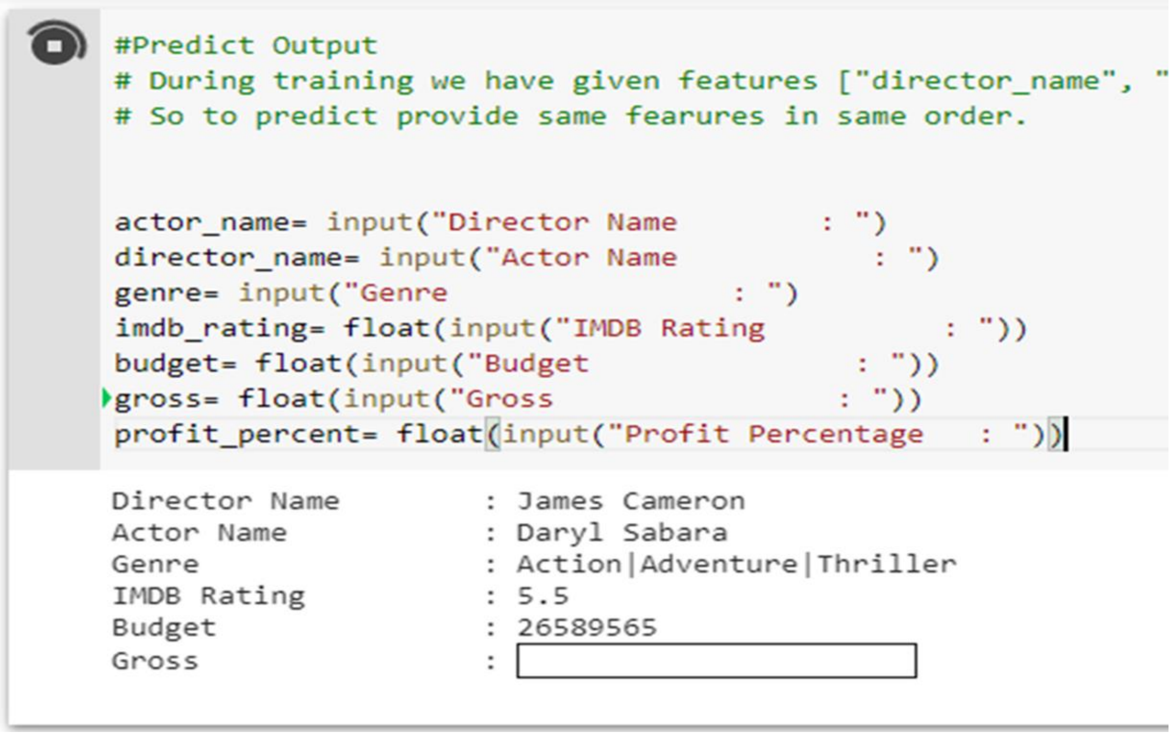
Finally, a user form was designed to accept user input from the user who wishes to carryout prediction. The fields include: Name, Genre, Director, Lead Cast, budget, revenue and IMDb rating. The user responds to the prompt to provide these inputs and he fills them out with the corresponding data of the movie you want to predict its success. These data can be gotten from the trailers of movies. For e.g. if a movie called "Sugar Tooth" was to be predicted, the data stated above have to be collected about the movie before its release date and entered into the prediction form, once this form is run, the prediction is carried out, and the result of the prediction is displayed as the class tag of the movie success, which ranges from blockbuster to horrible. This result is visualized in a very understandable format that does not require additional interpretation but can be understood by the user easily. This is shown in figure 7.



```
[ ]  df3.info()

     <class 'pandas.core.frame.DataFrame'>
     RangeIndex: 10866 entries, 0 to 10865
     Data columns (total 15 columns):
      #   Column                Non-Null Count  Dtype
     ---  ------                --------------  -----
      0   id                    10866 non-null  int64
      1   popularity            10866 non-null  float64
      2   budget                10866 non-null  int64
      3   revenue               10866 non-null  int64
      4   original_title        10866 non-null  object
      5   cast                  10790 non-null  object
      6   director              10822 non-null  object
      7   tagline               8042 non-null   object
      8   runtime               10866 non-null  int64
      9   genres                10843 non-null  object
      10  production_companies  9836 non-null   object
      11  release_date          10866 non-null  object
      12  vote_count            10866 non-null  int64
      13  vote_average          10866 non-null  float64
      14  release_year          10866 non-null  int64
     dtypes: float64(2), int64(6), object(7)
     memory usage: 1.2+ MB
```

Fig. 5: Dataset Information: Attributes for Prediction

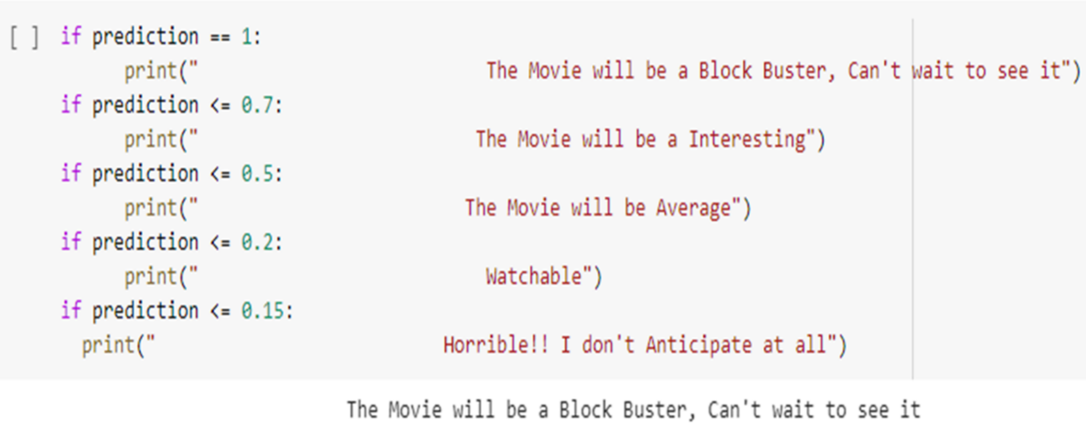


```
#Predict Output
# During training we have given features ["director_name", "
# So to predict provide same fearures in same order.


actor_name= input("Director Name        : ")
director_name= input("Actor Name          : ")
genre= input("Genre                 : ")
imdb_rating= float(input("IMDB Rating          : "))
budget= float(input("Budget                : "))
gross= float(input("Gross                 : "))
profit_percent= float(input("Profit Percentage    : "))
```

```
Director Name         : James Cameron
Actor Name            : Daryl Sabara
Genre                 : Action|Adventure|Thriller
IMDB Rating           : 5.5
Budget                : 26589565
Gross                 : 
```

Fig. 6: Movie Prediction Form



```
[ ] if prediction == 1:
          print("                          The Movie will be a Block Buster, Can't wait to see it")
    if prediction <= 0.7:
          print("                         The Movie will be a Interesting")
    if prediction <= 0.5:
          print("                        The Movie will be Average")
    if prediction <= 0.2:
          print("                          Watchable")
    if prediction <= 0.15:
      print("                          Horrible!! I don't Anticipate at all")
```

```
The Movie will be a Block Buster, Can't wait to see it
```

Fig. 7: Movie Prediction Output: Blockbuster

model efficiency, time complexity etc
No. of Training set = 80% of 27, 410
            Test set = 20% of 27, 410

Training Accuracy = Number of Correct Predictions * Prediction Error/100
Quality of Dataset = Training Error * 100 + Test Error * 100
The overall Efficiency of our model was gotten to be 97% while that of the existing system was 77%. This shows that our proposed model outperforms the existing model in accuracy and other parameters.

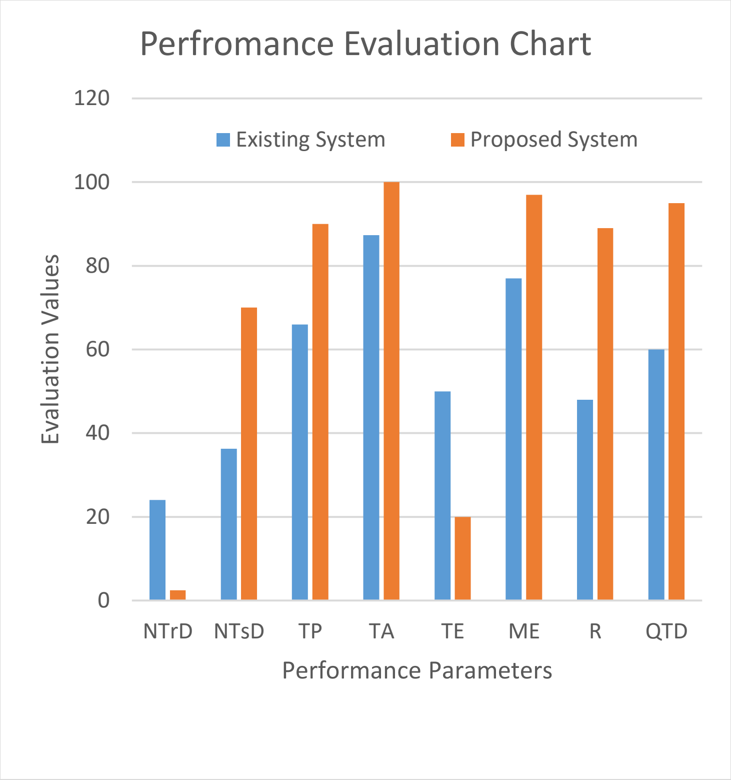


Fig. 8: Performance Evaluation Chart

S/N	Parameter	Existing Model	Proposed Model	Parameter
1	No. of Training Data (NTrD)	900	21, 926	No. of Training Data (NTrD)
2	No. of Testing Data (NTsD)	150	5448	Mean Square Error (MSE) (NTsD)
3	Training Speed (TP)	5 Minutes	1 Minute	Training Speed (TP)
4	Training Accuracy (TA)	87.3%	100%	Training Accuracy (TA)
5	Training Error (TE)	2e-01	1e-09	Training Error (TE)
6	Model Efficiency (ME)	77%	97%	Model Efficiency (ME)

7	Robustness (R)	48%	89%	Robustness (R)
8	Quality of Test Data (QTD)	60%	95%	Quality of Test Data (QTD)

2. Conclusion

Movie success prediction is a major problem of the stakeholders in movie production. His information is important to ensure informed decision without latter regrets resulting from gross losses of over or under investments. Data mining in PHP enables for data to be neatly scraped for data analysis and prediction and the Python programming language provides simplified data analysis (especially large datasets) and predictions. These tools were used to develop an enhanced movie success prediction which can accurately forecast the success rate of a movie before its final release. The viewers of the movie who pay subscription fees to see movies on special platforms will also benefit from this system in order to make informed decision on the kind of movies to subscribe for. The system has been tested using test sets and the prediction results were accurate. The system was also evaluated to ascertain its performance and it outperformed the existing system.

3. Contribution to Knowledge

An enhanced movie success prediction system using data mining PHP has been developed which outperforms the existing prediction system(s) in terms of accuracy and other parameters.

4. Suggestion for Future Work

Our future scope will be directed towards developing a model for predicting music success using machine learning algorithms. Another scope will be the use of a hybrid machine learning algorithm for the prediction of movie success.

VI. References

[1] P. Chakraborty, Z. Rahman and S. Rahman. "Movie Success Prediction using Historical and Current Data Mining." *International Journal of Computer Application*. Vol. 178, No. 47. PP. 1-5. Sep. 2019.

[2] P. Sanjai, J. Abhisht and M. A. Geetha. "Prediction of Movie Success for Real World Movie Datasets." *International Journal of Advance Research, Ideas and Innovations in Technology.* Vol. 3, Issue 3. PP. 455-461. 2017.

[3] A. Bhave, H. Kulkarni, V. Biramane and P. Kosamkar. "Role of Different Factors in Predicting Movie Success." 2015 International Conference on Pervasive Computing (ICPC). PP. 1-5.

[4] Apala, K. R., M. Jose., S. Motnam., C. C. Chan., J. Lizka and F. Gregorio. "Prediction of Movies Box Office Performance using Social Media." 2013 IEEE/ACM International Conference on Advances in Social Network Analysis and Mining. PP. 1209-1214. 2013

[5] Z. Mhowwala., A. R. Sulthana and S. D. Shetty. "Movies Rating Prediction using Ensemble Learning Algorithms." *International Journal of Advanced Computer Science & Application.* Vol. 11, No. 8. PP. 383-388. 2020.

[6] R. A. Suthana and S. Ramasamy. "Context Based Classification reviews using Association Rule Mining, Fuzzy Logic and Ontology." *Bulletin of Electrical Engineering & Information.* Vol. 6, No. 3. PP. 250-255. SEP. 2017.

[7] S. Rahim, A.Z.M.E. Chowdhury, A.I. Islam and R. Islam. "Mining Trailers Data from YouTube for Predicting Gross Income of Movies." *Research Gate Publication*. PP. 1-3. DEC. 2017.

[8] S. Wu., Y. Zheng, Z. Lai, F. Wu and C. Zhan. "Movie Box Office Prediction Based on Ensemble Learning." PP. 1-4. OCT. 2019

[9] A. Chauhan, D. Kumar and Ankit. “Movie Success Prediction using Data Mining.” *Research Gate Publication*. PP. 1-12. 2019.

[10] K. Meenakshi., G. Maragatham, N. Agarwal, I. Ghosh. “A Data Mining Technique for Analyzing and Predicting the Success of Movie.” IOP Conf. Series: *Journal of Physics*. Vol. 1000. PP. 1-10. 2018.

[11] Q. I. Mahmud, N. Z. Schchi, F. M. Tawsif, A. Mohaimen and A. Tasnim. “A Machine Learning Approach to Predict Movie Revenue Based on Pre-Released Movie Meta Data.” Journal of Computer Science. Vol. 16, Issue 6. PP. 749-767. JUN. 2020.

[12] A. Patil, A. Pujare, M. Shah and R. Barve. “Box office Movie prediction System.” *International Journal of Research in Engineering, Science and Management.* Vol. 2, Issue 4. PP. 88-91. APR. 2019.

[13] A. Upadhyay, N. Kamath, S. Shanghavi, T. Mandvikar and P. Wagh. “Movie Success Prediction using Data Mining.” *International Journal of Engineering Development & Research.* Vol. 6, Issue 4. PP. 198-203. 2018

[14] G. S. Brar and A. Sharma. “Sentiment Analysis of Movie Review using Supervised Machine Learning Techniques.” *International Journal of Applied Engineering Research*. Vol. 13, No. 6. PP. 12788-12791. NOV. 2018

[15] S. Caghyor, B. Oztaysi and S. Sezgin. “Forecasting Box office Performances using Machine Learning algorithms.” *Springer.* PP. 257-264. JAN. 2020

[16] J. M. Kim, L. Xia, I. Xia, S. lee and K. H. Lee. “Finding Nemo: Predicting Movie Performances by Machine Learning Methods.” *Journal of Risk and Financial Management.* Vol. 13, No. 93. MAY. 2020.

[17] J. Ponce, A. Hernandez, A. Ochoa, F. Padilla, F. Alvarez and E.P. de Leon. “Data Mining in Web Applications.” Data Mining & Knowledge Discovery in Real Life Applications. PP. 438. FEB. 2009.

[18] B. M. Ramageri. “Data Mining Techniques & Applications.” *Indian Journal of Computer Science and Engineering.* Vol. 1, No. 4. PP. 301-305.