Journal of Computer Science

Original Research Paper

A Machine Learning Approach to Predict Movie Revenue Based on Pre-Released Movie Metadata

[1]Quazi Ishtiaque Mahmud, [2]Nuren Zabin Shuchi,
[1]Fazle Mohammed Tawsif, [3]Asif Mohaimen and [3]Ayesha Tasnim

[1]Institute of Information and Communication Technology, Shahjalal University of Science and Technology, Sylhet, Bangladesh
[2]Electrical and Electronic Engineering, Shahjalal University of Science and Technology, Sylhet, Bangladesh
[3]Computer Science and Engineering, Shahjalal University of Science and Technology, Sylhet, Bangladesh

Article history
Received: 17-03-2020
Revised: 21-05-2020
Accepted: 09-06-2020

Corresponding Author:
Quazi Ishtiaque Mahmud
Institute of Information and Communication Technology,
Shahjalal University of Science and Technology, Bangladesh
Email: rafisustcse@gmail.com

Abstract: With the growth of the movie industry, it is becoming increasingly important for the stakeholders to get an idea about the probable profit made by the movie in the box office. In fact, among movies produced between 2000 and 2010 in the United States, only 36% had box office revenues higher than their production budgets, which further highlights the importance of making the right investment decisions. To address this issue, different machine learning algorithms like Logistic Regression, Support Vector Machine (SVM) and Multi Layer Perceptron (MLP) are used in this study to predict the box office return of a movie based on the data available before the release of the movie. The models use 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. An analysis of different machine learning architectures is also presented in this research. Finally, a system is proposed that produces comparable results with existing researches in this field and it can predict the profit generated by a movie with a "one class away" accuracy of 85.31% without using any sales information.

Keywords: Continuous-Valued Features, Binary Features, Logistic Regression, Support Vector Machine, Linear Kernel, KNN, Polynomial Kernel, RBF Kernel, Multi Layer Perceptron, Activation Functions

Introduction

The movie industry is one of the first forms of industrialized mass-entertainment and has exhibited remarkable growth in the last few decades bringing about a huge revenue for its stakeholders. Making sure that the revenue generated by a film reaches above the cost of making the movie has always been a prime concern for its investors. For example, A Korean film named "Mr. Go" (2013) was estimated to generate a revenue of around 20 million dollars by reaching out to at least 5 million users but ended up reaching only 1.5 million resulting in huge disappointments for the investors (Lee *et al*., 2018). A system that estimates the box office return of a movie can be a useful tool for the stakeholders in making informed financial decisions and adjusting the marketing strategy to increase the probability of success.

Kim *et al*. (2013a), Chikersal *et al*. (2015), Asur and Huberman (2010) used user reviews and comments to predict the success of a movie. They used Support Vector Machine to classify the user comments between positive and negative classes. Then predicted the success of movies based on the number of positive and negative reviews. Other architectures were also used in different researches. KNN classifier was used by Alsaffar and Omar (2015) for Malay movie reviews, a hybrid model consisting of Multi Layer Perceptron (MLP) and Naïve Bayes (NB) was proposed by Al-Batah *et al.* (2018) for Arabic movie reviews, SVM applied genetics (GSVM) and KNN classifier was also used by Mohamed *et al.* (2018) for the "Cornell Movie Review Dataset (polarity dataset v2.0)" (2004). Pennock (2000), also illustrated that online activities such as user comments indeed play a part in determining the financial outcome of artificial markets.

Also, some studies tried to illustrate the importance of other factors such as violence and horror, in determining the fate of a movie (Gunter, 2018). Also, an attempt has been made to analyze the effects of features such as sequels, number of initial screens, comments regarding a film, presence of stars in films to determine the ultimate fate of the movie (Lee *et al*., 2018).

But apart from these factors, there may be other factors such as production house, director, social reach of the cast, preference of the audience, duration, number of users for reviews, language, country, content rating, budget, title, year and release date that can affect the profit generated by a movie. In this literature, we tried focusing on these factors as data related to many of these factors are available before the release of a movie. We wanted to explore how these features contribute to the final revenue that is generated by a film. Also one of our main goals was to be able to predict financial success in the very early stages. Most of the features mentioned above, for example, the production house, social networking profile of actor/actress, director, budget, genre etc., can be obtained even before the release of movie trailers. So, an attempt has been made in this research to develop a system that will be able to provide us with enough insights about the movie's performance in the box office by using the parameters described above.

Considering the pre-release data provides manifold advantages. The marketing and advertising strategies can be adjusted based on the prediction by the system. Again, since the data related to cast, director, plot, social reach of the actors are available right after making the decision, the early prediction can help in making adjustments in these parameters to increase the probability of success in the box office.

The paper is organized in the following order. Section two discusses the previous works done related to predicting the success of a movie. The third section describes the methodology used to design the model which includes data collection, preprocessing of data, feature extraction, the definition of the class labels and analyzing the performance of different classifiers. Finally, in section four an overview of the developed system is given and scopes for future improvements are discussed.

Previous Works

Some attempt has been made to predict the financial success of a movie. Sawhney and Eliashberg (1996) found that by using not more than two data points it is possible to predict box office revenues with good accuracy. But when they tried to guess box office revenues without sales information they could not achieve good accuracy.

Litman and Kohl (1989) tried showing in their research that movies having won academy awards and having superstars will be more likely to succeed in the box office.

Ravid (1999) argued in their research by saying that not only movies that have a lot of superstars involved but also movies that have a huge budget is more likely to succeed. They also found that Sequels and Family movies contribute more to the overall success of the film than other features.

Simonoff and Sparrow (2000) found a relationship between winning the Oscar award and generating revenue. They also claimed, if the same types of movies are released at the same time then there is a possibility that their revenue might fall. It was also reported that the first few weeks' earnings play a vital role in determining the fate of the film.

Brewer *et al*. (2009) divided the information regarding movies into two stages: pre-release and post-release. According to their research, the factors that play an important role in the pre-release phase of a movie are budget, sequel and time of movie release. In the case of the post-release phase, the number of screens, the presence of stars and nominations for awards play a vital role in generating revenue for a film.

Chang and Ki (2005) also showed that sequels, the number of first week screens and movie release time affect the revenue generation process.

Kim *et al*. (2013b) pointed out that the critics' reviews and the online discussion of general people play an important part in determining the chances of success of a film.

Bayesian model was also used to generate predictions on movie viewership (Neelamegham and Chintagunta, 1999). They found that among all the features the number of screens plays the most vital role.

Zufryden (1996) analyzed the effects of advertising and the intensities of the theatre distribution on the success of a film.

Treme (2010) analyzed the effects of media exposure of a celebrity in the magazine, "People" (2020). They found that in most cases the media exposure of the celebrity involved in the movie plays a vital role in determining the fate of the film rather than promotional advertisements before the release of the film.

Among other works, Litman (1983) used decision support systems, then another study showed that the first two weeks of screening are most important which contributes to 25% of the total revenue (Litman and Ahn, 1998). Sochay (1994) tried to predict the success of movies by analyzing initial theatrical release.

Abel *et al*. (2010) used Naïve Bayes and Support Vector Machine (SVM) classifier to predict the financial outcome of movies and music. They reported that SVM performed better in their case. In their research, the number of appearances of the movie title in social blogs was primarily chosen as the feature. They reported that their model achieved a precision score of 47.73 and 59.77 for the Naïve Bayes and the SVM model respectively.

Kim *et al*. (2015) used Genetic Algorithms for feature selection. Then they experimented with Multivariate Linear Regression, Support Vector Regression (SVR) and KNN. They tested with only 212 Korean films and also used cross-validation as they don't have a large dataset. They divided their dataset into three portions and collected the target variables (revenues) for one week after, two weeks after and three weeks after the release of the film. They reported that they achieved

Root Mean Squared Error (RMSE) score of 0.0262, 0.0463 and 0.0195 for the test revenues of one, two and three weeks respectively. So, they only considered predicting the first three weeks of revenue data rather than predicting the whole revenue of a film.

Rhee and Zulkernine (2016) tried to predict movie profitability by using Neural Network along with movie metadata and social media data. However, they only considered two classes. The revenue generated was divided by 2 and then subtracted from the budget to find the profit. For positive profit, the movie was considered as ‘successful’ and for negative profit, the movie was considered as ‘flop’. So, they did not find the revenue of the movie. They only tried to predict whether the movie was financially successful or not. They used 375 movies to design their model. They tested their system for only 56 movies and reported an accuracy of 88.8%.

Lash and Zhao (2016) divided the whole feature space into 3 types: “Who”, “What” and “When”. Where “Who” means the actors, directors and the casts who are involved in the movie. “What” represents the genre and MPAA ratings and “When” represents the time of the release of the film. They ran their model using 2506 movies. As their dataset is small, they implemented the cross-validation technique. They used Logistic Regression as their prediction model and achieved around 80% accuracy. But they used only 3 classes: “success”, “failure” and “average” movies.

Choudhery and Leung (2017) used Polynomial regression to predict movie revenues. They mainly focused on the social media features while conducting their experiment. For each movie, only three features were considered in the research number of tweets, number of positive tweets and number of negative tweets. Only six movies were considered for evaluating the model. They reported a Mean Squared Error (MSE) of about 13%. It was mentioned in the research that the best performance is achieved by applying 6th degree polynomial regression function.

Shim and Pourhomayoun (2017) tried to predict the opening weekend revenue with a Linear Regression model. They collected Twitter data from 67 movies and they considered the features number of tweets, number of positive and negative tweets, presence of special characters known as ‘emojis’ in tweets, number of theatres, budget and the weather condition of the opening weekend of the movie. An accuracy of 65% was reported in their research. They improved their prediction error from 35 to 31% by creating separate clusters of the dataset and then applying Linear Regression on the separate clusters.

Ahmad *et al.* (2017) tried to predict the rating rather than predicting the revenue based on Bollywood movie metadata. They considered the higher-rated films as the most successful films. Their feature set includes Year, Director, Producer, Genre and Language. They predicted the rating of the movie based on the correlation of the features. In the research, it was mentioned that Genre plays an important role in predicting the rating of a film. They also found a strong correlation between Actors and Genres of a film.

Quader *et al.* (2017) tried to predict the revenue generated by a movie by using the movie metadata. They considered both pre-released and post-released data. However, they did not consider the features ‘Genre’ and ‘Sequel’. Also, they only considered the director’s popularity as a feature. But, the popularity of the actor/actress in a movie was not considered in their experiment. Also, the language was not considered in their feature set. It is reported that 15 features are used and the movies are divided into five classes (from ‘Flop’ to ‘Blockbuster’). They ran their experiment for 755 movies and reported a bingo and one class away accuracy of 58.5 and 89.67% respectively by using a MLP model. As they had only 755 movies, 10-fold cross-validation was used while performing the experiments.

Xiao *et al.* (2017) used Linear Regression to predict revenue generated by movies in China. In their research, authors mainly focused on the Genre and Cast Popularity of the movie. The features that they worked with are Investment, Title, Script, Time length, Schedule, Rival, Genre, Cast, Award and Advertising. However, it was mentioned in the research that the features: Investment, Script, Time Length and Awards play little role in determining the revenue generated by the film. “Mean Absolute Percentage Error” (MAPE) score was used to evaluate the performance. They collected their data from five video websites. But, MAPE score of only two of the sources was shown. They considered predicting the first week revenue (fr) and the overall revenue (or). The best MAPE score (or: 1.2042%, fr: 1.4259%) is achieved for the data of 14 movies extracted from the video website “Youku”.

Sachdev *et al.* (2018) used Linear Regression and Decision Tree regression for predicting movie revenue. However, they claimed to achieve better accuracy using the Decision Tree regression model. They divided their training sample into 3 sections on the values of the features ‘Number of Screens’ and ‘Tomato Rating’ (average of critics rating) and ran regressions separately on each set. They tested their model for 400 movies and reported an error rate of 24.76%.

Different regression algorithms are also used to predict movie revenue from movie metadata (Walanaraya *et al.*, 2018). In their research, they used 10 features from each movie. They considered the following features: Budget, Vote count, Vote average, Runtime, Genres, Spoken Language, Production companies, Release date and Casts. However, the features like the content rating of the film, the release country, the popularity of the film in social networking sites were not considered. Along with this feature set, three different regression algorithms Linear, Polynomial and Support Vector Regression (SVR) were applied. They reported that Linear Regression outperformed the other two architectures. They claimed that the best R^2 value (39.73%) was achieved for 4 clusters by using Linear Regression.

Galvão and Henriques (2018) tried predicting the movie revenue by using regression, decision trees and neural networks. They worked with 1920 movies. Their feature-set included: Sequel, MPAA rating, Genre, Budget, Oscars, Awards, Directors, Actors, Season, Spectators and Critics. A total of 9 classes were used. It was reported in the research that MLP with 3 hidden layer neurons performed the best. They used 70% of their data for training and 30% of their data for testing. Out of 295 observations, it was reported that their model could correctly predict the class of 39% of the samples.

During our study, it is observed that not a lot of researches are carried out to predict box office revenue at the earliest stage possible by using the pre-released movie metadata. Some of the researches that were carried out used a very small amount of data to perform their experiments. Again, some works have been done to predict movie revenue based on initial sales data, some researches were done to predict the rating of the movie, some works were carried out to predict the profitability of a movie. So, we decided to analyze the effectiveness of different machine learning algorithms to solve the problem of movie revenue prediction based on pre-released movie metadata without any sales information because sales information is not available prior to the release of the film.

Methodology

In this section, a description of our prediction system is given. Figure 1 provides an overview of the whole system. Each component of the system is explained in the subsequent subsections.

Data Collection

In this research, the "IMDB 5000 Movie Dataset" (2020) is used. This dataset contains 5000+ movie metadata scraped from IMDB. It contains 28 variables for 5000+ movies, spanning across 100 years in 66 countries. Also, there are 2399 unique director names. The variables are movie title, number of critics for reviews, movie's facebook likes, duration, director's name, director's facebook likes, actor 3's name, Actor 3's facebook likes, actor 2's name, actor 2's facebook likes, actor 1's name, actor 1's facebook likes, gross (revenue), genre, number of voted users, cast's total facebook likes, number of faces on poster, plot keywords, movie's IMDB link, number of user for reviews, language, country, content rating, budget, title, year, IMDB score and aspect ratio.

Data Preprocessing

The dataset contains data in csv format. It needed some preprocessing. As mentioned earlier the dataset contained data of over 5000 movies and each movie has 28 variables. Each of these variables can be used as a feature. But while processing data some of the values of these variables are found to be missing. So, those movies are removed from our dataset. Also, there were some invalid data. Those movies are removed too. Finally, 3200 movies are chosen. 70% (2240 movies) of the dataset are used to train our machine learning models and 30% (960 movies) of the dataset are used to test our models.

Feature Extraction

As mentioned earlier, the dataset has 28 variables for each movie. These 28 variables are converted into 35 features for each movie. Some of those features have continuous values and some of them are binary features (their value is either 0 or 1). Brief descriptions of all the features are discussed here.

Continuous-Valued Features

The continuous-valued features that are considered is discussed below:

- Director's facebook likes, contains the total number of facebook likes of the facebook page of the director of the movie. All the values are integers within the range of 2 to 23000
- Number of critics for reviews, this feature represents the number of critical reviews that the movie received. The values of this feature are also integers within range 1 to 813
- Actor 3's facebook likes, contains the number of likes the 3rd actor/actress has on his/her facebook page. This is also an integer-valued feature within range 2 to 23000
- Actor 2's facebook likes, holds the total facebook likes of the 2nd actor/actress of the movie. The feature values are integers in the range 2 to 137000
- Actor 1's facebook likes, this feature reflects the popularity of the main actor/actress of the movie. It represents the total facebook likes of the leading actor/actress of the movie. This is also an integer-valued feature. The values are within range 2 to 640000 with almost 75% of the actors/actresses having 10000 facebook likes or higher

- Movie's facebook likes, represents the total number of likes of the facebook page of the movie. Also, an integer-valued feature with values within the range 2 to 349000
- Duration, indicates the length of the film. The values are integers. The lengths are expressed in minutes. The shortest film has a duration of 7 min while the longest movie has a duration of 511 min
- Number of voted users, means the number of users who voted for the movie. Also, an integer-valued feature with values between 5 to 1690000
- Reviews, reflects the number of reviews that were written by reviewers for the film. This feature has integer values within range 1 to 5060
- Budget, means the overall cost for the production of the film. The values of budget are also integers. The lowest budget movie has a cost of 218 dollars. Whereas, the costliest movie has a budget of 12.2 billion dollars
- IMDB score, points to the IMDB score of the movie before the release of the film. This feature has decimal values. The lowest score is 1.6 and the highest score is 9.5
- Total facebook likes, this feature represents the total facebook likes of all the casts of the movie. The values are integers within range 2 to 657000

As these features have very different values we used normalization technique to normalize the feature values. For normalization, we divided each of the feature values with the highest value of that feature. So after normalization, each of the features will have a value between 0 and 1.

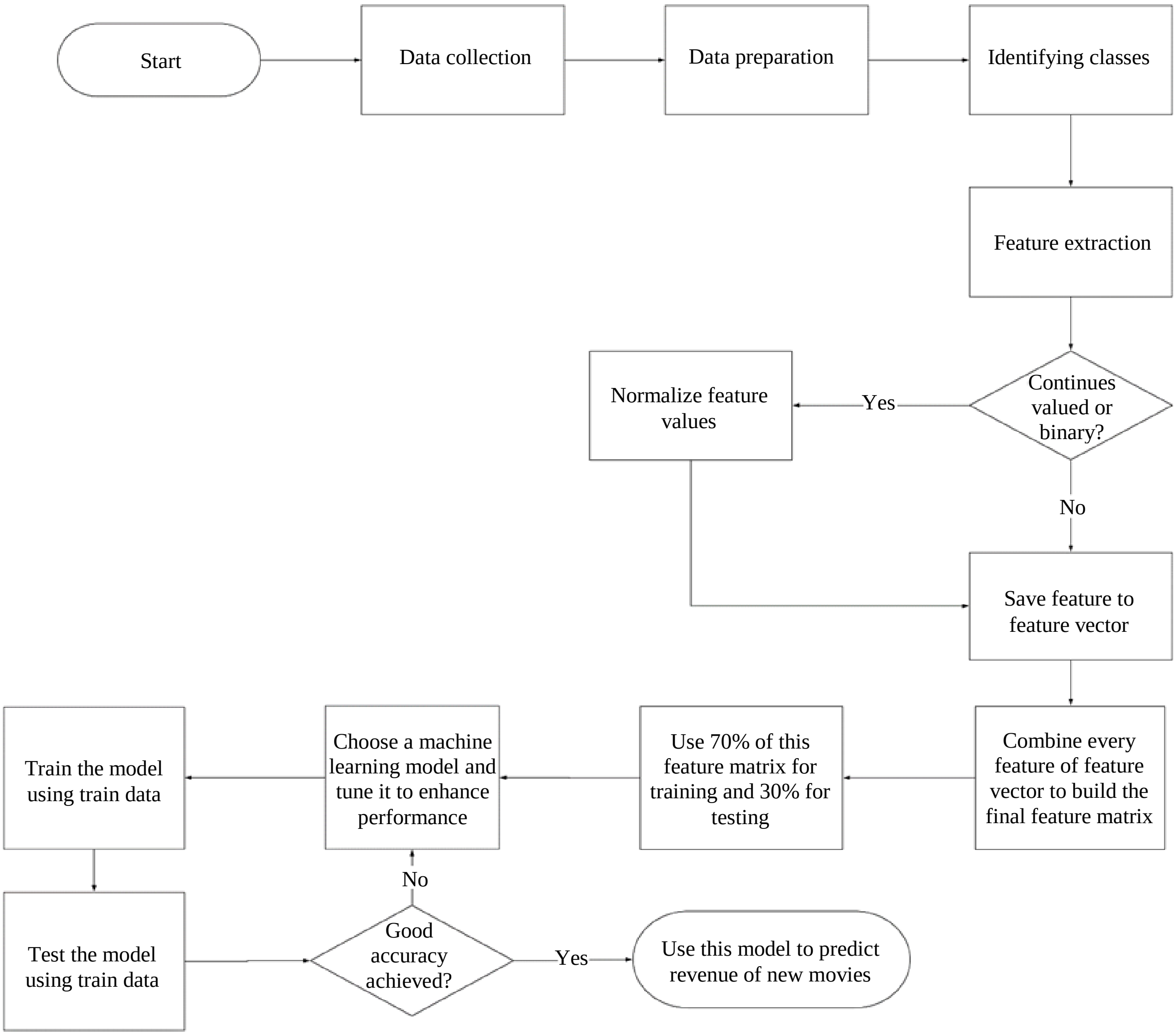
Start
Data collection
Data preparation
Identifying classes
Feature extraction
Continues valued or binary?
Yes
Normalize feature values
No
Save feature to feature vector
Combine every feature of feature vector to build the final feature matrix
Use 70% of this feature matrix for training and 30% for testing
Choose a machine learning model and tune it to enhance performance
Train the model using train data
Test the model using train data
Good accuracy achieved?
No
Yes
Use this model to predict revenue of new movies

Fig. 1: Our methodology

Table 1: Class labels

Class	Revenue (in millions)
0 (flop)	< 1
1	> 1 and < 10
2	> 10 and < 20
3	> 20 and < 40
4	> 40 and < 65
5	> 65 and < 100
6	> 100 and < 150
7	> 150 and < 200
8 (blockbuster)	> 200

Binary Features

In this subsection, considered binary features are discussed.

Genre

We considered the genre of a movie. There are 17 genres: Action, Adventure, Fantasy, Sci-fi, Thriller, Comedy, Family, Horror, Animation, Romance, Musical, Documentary, Drama, History, Biography, Mystery and Crime. Whenever a movie falls into any type of genre we set the position of the genre to 1, otherwise it is 0. For example, if a movie has the following genre Action, Adventure, Mystery the feature value will be like this, [1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0] where each position in the vector indicates the genres that are described before.

Content Rating

Content rating is also considered as a binary feature. But it only refers to a certain category. So the feature will be like this for Parental Guidance Suggested, [0, 1, 0, 0, 0] where each position in the vector represents the content ratings: General Audiences, Parental Guidance Suggested, Parents Strongly Cautioned, Restricted, Adults Only.

Language

The Language is also considered as a feature of the movie. If it is English then the value 1 is used, for any other language 0 is used.

Defining Class Labels

As the goal is to predict revenues, there needs to be a way to divide continuous values into some discrete classes. The revenue generated by a movie is collected from the 'Gross' feature of "IMDB 5000 Movie Dataset" (2020). Nine categories (0 to 8) of revenues are considered in this research as suggested by Sharda and Delen (2006). The information about the revenue classes is discussed in Table 1. Here a movie that made less than 1 million is considered to be a flop and a movie that generated more than 200 million is considered as a blockbuster film.

Experimenting and Analyzing Performance of Different Classifiers

Here the description of the classifiers that are used for our classification problem and analysis of the performance of the models are provided. For all the models "PSMLL" (2020) is used.

Performance Evaluation Metrics

To measure the performance of our models two metrics are considered: Bingo accuracy and one class away accuracy. Bingo accuracy means that the classifier correctly predicted the class of a movie. One class away accuracy means the classifier predicted the class of a movie within one class distance. For example, if a movie belongs to class 4 then if our classifier predicts the class to be 3 or 5 then according to one class away accuracy metrics, it successfully predicted the class of that movie.

Experimenting with Logistic Regression

Logistic Regression is used for solving categorization problems. Firstly, it generates a hypothesis by using the following formula:

$$z = \vec{w}.\vec{x} + b \tag{1}$$

Here, z is the hypothesis value, $\vec{w}$ represents the weights, $\vec{x}$ represents the features. They both are vectors. b represents the bias. The value of z is passed through a sigmoid function shown below which converts this value into either 0 or 1:

$$sigmoid\left(z\right) = \frac{1}{1+e^{-2}} \tag{2}$$

So, it works as a binary classifier. For more than two classes it first calculates the probability of belonging to the first class and the rest of the class. Then it calculates the probability of belonging to the second class and the rest of the class. It calculates the probability for each class like this and finally selects the class with the highest probability. This method is known as the one vs rest methodology. The value of the regularization parameter is chosen as $1*10^{-5}$. This helps to prevent overfitting. After designing this classifier, the model is tested. As the dataset is not huge 5-fold cross-validation technique is used. For each iteration, 70% data are considered as training samples and 30% data are kept for testing. A maximum bingo accuracy of 55.93% is achieved using the Logistic Regression and if one class away accuracy is considered, the accuracy rises to 82.81%. Figure 2 represents the findings.

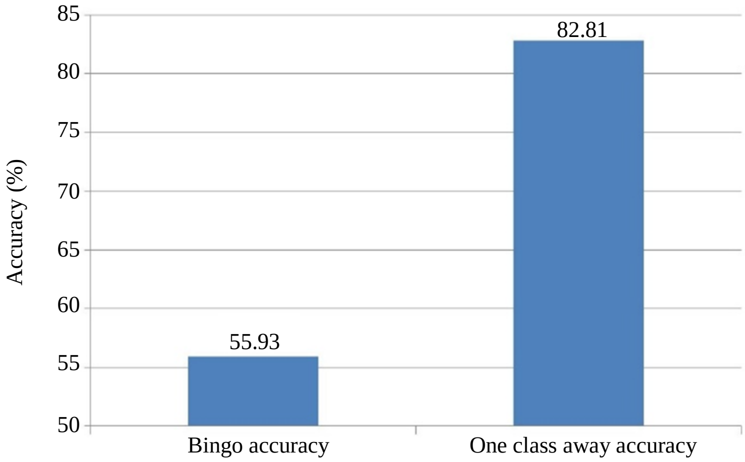

Fig. 2: Bingo and one class away accuracy for Logistic Regression model

Experimenting with Decision Tree

CART decision tree is also implemented to classify among the movies. It can be used for both classification and regression problems. It works in a recursive manner. At each step based on an impurity function, a feature is selected such that the impurity is minimized. Then a threshold feature value is used to divide the dataset into two smaller subsets. This process continues unless the data points that belong to the same class are grouped together. At each step, a new feature with a new threshold is chosen. The impurity function depends on whether the problem is a classification or regression problem. Now, as ours is a classification problem we used the Gini index as the impurity function which uses the following equation to calculate the impurity of a feature:

$$Gini\ index = 1 - \sum_{i=1}^{Number\ of\ classes} P_i^2 \tag{3}$$

Here, P_i represents the probability that a sample data point belongs to a particular class given a specific value for the feature. Figure 3 represents our findings. As can be seen, CART decision tree based approach achieved a bingo accuracy and one class away accuracy of 50.1% and 79.37% respectively.

Experimenting with KNN

KNN can also be used for classification and regression tasks. But we defined our problem as a classification problem. For a query data point, KNN finds out the K nearest data points and assigns the class that is the most frequent among these neighbors. Euclidean distance is used to find the K nearest neighbors. We experimented with different values of K and found the best accuracy (bingo: 54.17%, one class away: 82.81%) is achieved for k = 7. Figure 4 represents our findings.

We also tried assigning more weights to the closer neighbors than the distant neighbors so that they contribute more in determining the class of the query point. But we found that assigning more weights to the closest neighbors did not increase performance in our case. Here the weight is considered as the inverse of the distance of the neighbor from the test data point. Figure 5 illustrates the findings.

Experimenting with Support Vector Machine (SVM)

Support Vector Machine is a very commonly used binary classifier. To classify among more than two classes it also uses the one vs rest methodology. It uses the following equation to classify among data points:

$$f(\vec{x}) = sign\left(\vec{w}^t.\vec{x} + b\right) \tag{4}$$

Here $\vec{w}^t$ and $\vec{x}$ both are vectors. $\vec{w}^t$ represents the learned weight of each of the features and $\vec{x}$ represents the feature vector. So, it calculates the dot product between the weight and feature vector. SVM creates an optimal hyperplane to classify among data points and it calculates the signed distance between a data point and the hyperplane. The sign of the distance determines to which class the data point belongs. SVM uses kernel trick to classify dataset that contains non-linear characteristics. Kernel function converts lower dimensional input space to higher dimensional input space. For example, let us take a simple kernel function $f(x) = x^2$. Figure 6 shows a sample dataset. It can be seen that it is not linearly separable. Suppose, the feature vector is $X1$. So, $X1$ = [[2], [3], [4]]. Now, if $X1$ is passed through our kernel function the feature vector will look like this, $X1$ = [[2, 4], [3, 9], [4, 16]]. So it can be observed that earlier each of the data points had only one dimension (one feature) but now every data point has two dimensions (two features). Figure 7 represents the fact that now the data points are linearly separable. So kernel function increases the dimension of the feature vector so that linearly inseparable datasets can be separated linearly.

All three of the kernels (Linear, Polynomial and RBF) are implemented to measure performance.

Firstly, an experiment with Linear kernel is performed. Linear kernel does not increase the dimension of the feature vector. It works directly with the given features. Figure 8 shows that Linear kernel gives us 54.69% bingo and 83.23% one class away accuracy. Here also 5-fold cross-validation is applied.

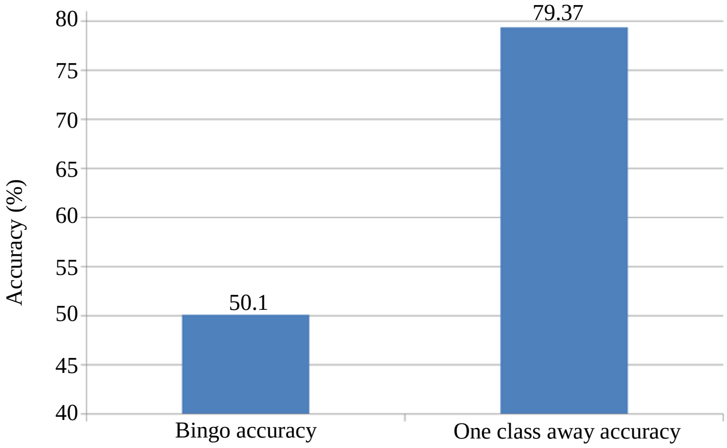

Fig. 3: Bingo and one class away accuracy for Decision Tree model

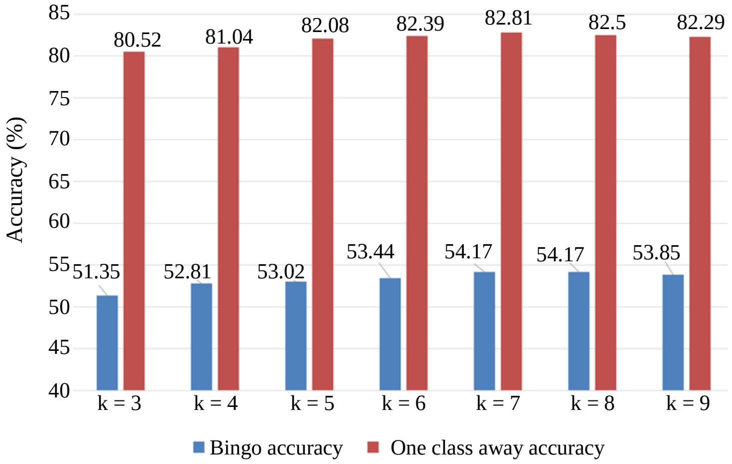

Fig. 4: Bingo and one class away accuracy for KNN classifier (with uniform weight distribution)

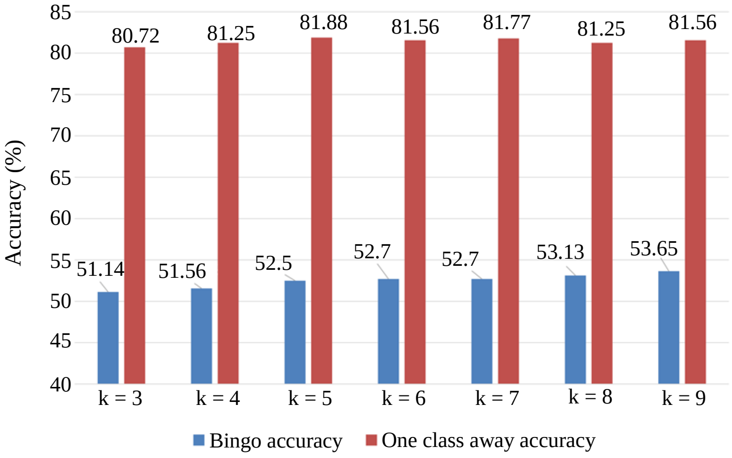

Fig. 5: Bingo and one class away accuracy for KNN classifier (with varied weight distribution)

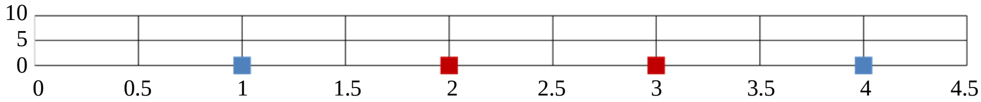

Fig. 6: Linearly inseparable dataset

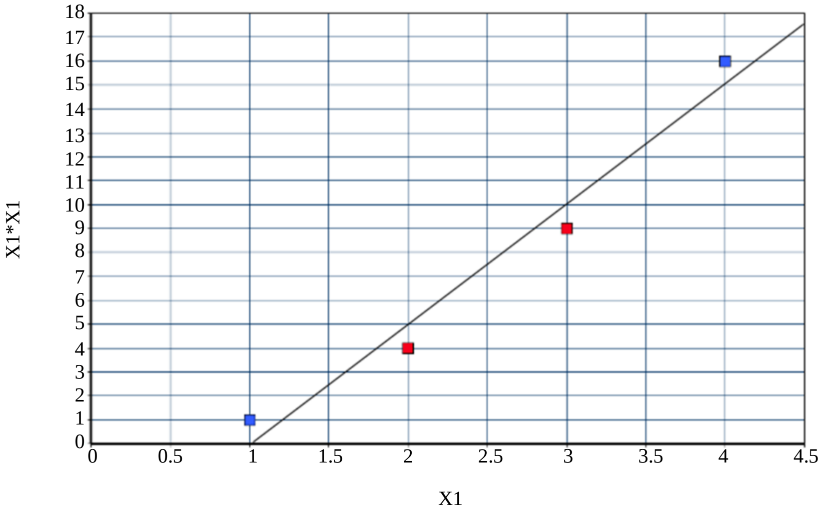

Fig. 7: Linearly separable dataset (after converting to 2D)

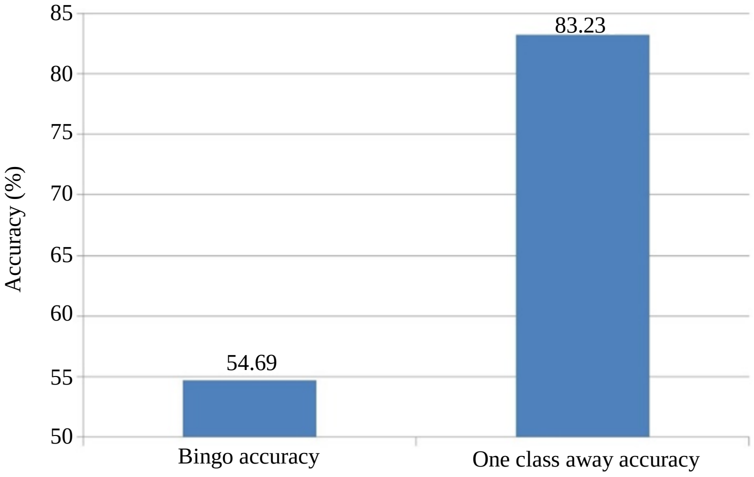

Fig. 8: Bingo and one class away accuracy for SVM with Linear kernel

Then, the Polynomial kernel is implemented. Polynomial kernel uses the following equation to convert the data points to a higher dimension:

$$k\left(\vec{x}_i,\vec{x}\right)=\left(1+\vec{x}_i^t\vec{x}\right)^d \tag{5}$$

Here, $\vec{x}_i$ means the feature vector and $\vec{x}$ means the test data point which is also a vector. Here d = 1 means that the data dimensions are not increased. So there is no difference between a linear kernel and a Polynomial kernel where d = 1. When d = 2, it means that not only single features but also a pair of features are considered. For example, the two features 'Director's Facebook Likes' and 'Number of Critics for Reviews' can be a feature pair together. Again, 'Content Rating' and 'Language of the Film' these two can be a feature pair. When d = 3 feature triplets are considered. This is how Polynomial kernel increases features by introducing new features by combining the existing features. Figure 9 represents accuracy for Polynomial kernel and it can be seen that it has 49.79% bingo and 79.16% one class away accuracy (5-fold cross-validation used).

Finally, RBF kernel is implemented which takes on the following form:

$$k\left(\vec{x}_i, \vec{x}\right) = \exp\left(-\gamma \left\|\vec{x}_i - \vec{x}\right\|^2\right) \quad (6)$$

Here, $\vec{x}_i$ means the feature vector and $\vec{x}$ means the test data point which is also a vector. γ represents the slop between them. Figure 10 represents the bingo and one class away accuracy for RBF kernel after implementing 5-fold cross-validation.

From Fig. 8 to 10 it can be seen that the bingo accuracies for Linear, Polynomial and RBF kernel are respectively 54.69%, 49.79% and 51.88%. But better accuracy is achieved if one class away accuracy is considered. For Linear, Polynomial and RBF kernel 83.23%, 79.16% and 80.42% accuracies are achieved respectively. So, it can be observed that Linear kernel produced the best results. That indicates, increasing dimension of the feature space is not useful in determining the profit of a movie.

Experimenting with Support Vector Regression (SVR)

In SVR, it is possible to control the amount of error that is allowed. Here the main goal is to minimize the following objective function:

$$\frac{1}{2}\left\|w\right\| + C * \sum_{i=1}^{number\ of\ samples} \xi_i$$

Here w is the weight vector, C is a constant, ξ is a slack variable which indicates the deviation of each sample data point from the allowed error margin (ε). Also, the following constraints need to be fulfilled:

$$|y - wx| \le \varepsilon + |\xi_i|$$

x is the test data point vector, ε is the allowed error margin, w and ξ_i are same as the objective function. The value of C can be tuned to increase or decrease tolerance for data points that are outside of the allowed error margin (ε). Linear, Polynomial and RBF kernel discussed previously can also be used with SVR. Figure 11 shows our findings. It is observed that if bingo accuracy is considered then SVR with linear kernel performs the best (56.25% accuracy) but if one class away accuracy is considered then SVR with RBF kernel performs the best (83.95% accuracy). In all cases, we considered C = 100 and ε = 0.1.

Experimenting with Multi Layer Perceptron (MLP) Model

Multilayer Perceptron model is very popular nowadays for solving the problem of classification and also for predicting non-linear functions. In MLP the number of neurons in the input layer is the same as the number of features. The hidden layer normally contains half of the number of input layer neurons (Marsland, 2014). The number of neurons in the output layer is equal to the number of classes. In this research, the profit generated by a movie is distributed among 9 classes. So, there are 9 neurons in the output layer. Our total feature matrix consists of 3200 rows and 35 columns. So, there are 3200 movies and 35 features and our neural network will have 35 neurons in the input layer.

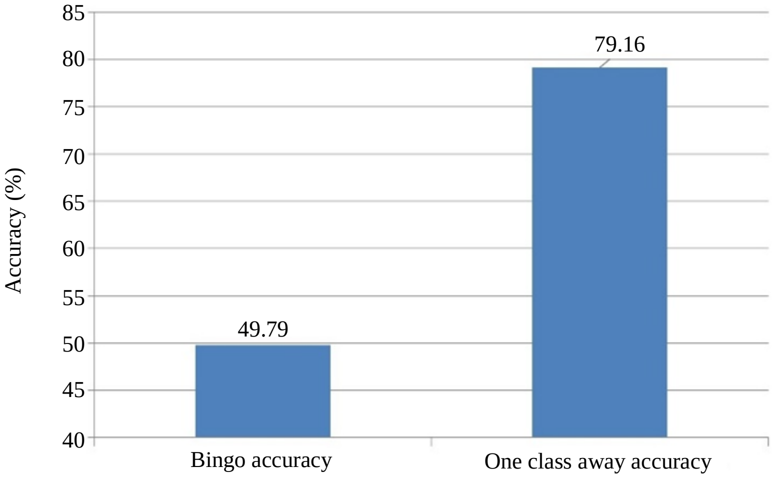

Fig. 9: Bingo and one class away accuracy for SVM with polynomial kernel

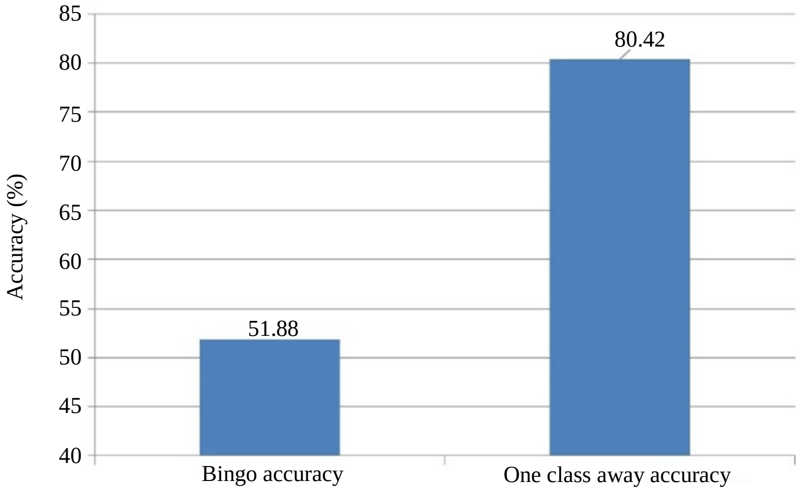

Fig. 10: Bingo and one class away accuracy for SVM with RBF kernel

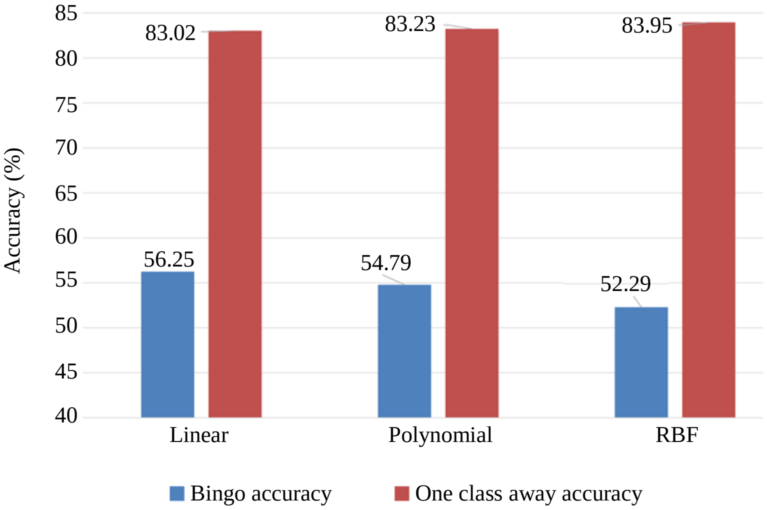

Fig. 11: Bingo and one class away accuracy for SVR with linear, polynomial and RBF kernel

Figure 12 represents the neural network architecture. It can be seen that the input layer has 13 “blue” neurons and 2 “orange” neurons. The “orange” neurons contain hidden neurons. For example, the neuron representing the feature “Genre” contains 17 hidden neurons for 17 genres and, the neuron representing the feature “Content rating” contains 5 hidden neurons for the 5 content rating types. Figure 13 and 14 represent an illustration of the state of the neuron that contains the features “Genre” and “Content rating” of a movie respectively.

One of the most important steps in implementing a MLP model is the initialization of the weights. The weights are initialized between values $-\frac{1}{\sqrt{n}}$ and $\frac{1}{\sqrt{n}}$ (Marsland, 2014). Here *n* represents the number of input layer neurons.

There exist four different types of activation functions: Identity, logistic, tanh and relu. Activation function defines what will be the output value of the neurons at the output layer. Experiments are carried out for different activation functions. The identity activation function has the following form:

$$f(x) = x \qquad (7)$$

It’s the simplest activation function. That means the neuron will output whatever value it has. Now the problem with the identity activation function is that it will not perform well if the dataset contains nonlinear characteristics.

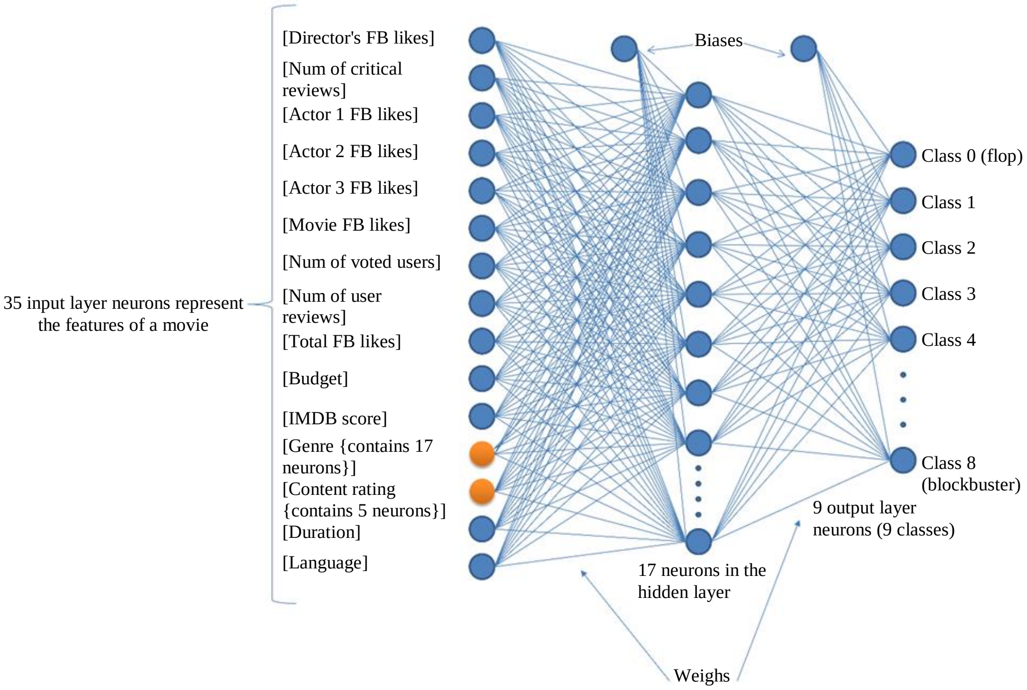

Fig. 12: Our neural network architecture

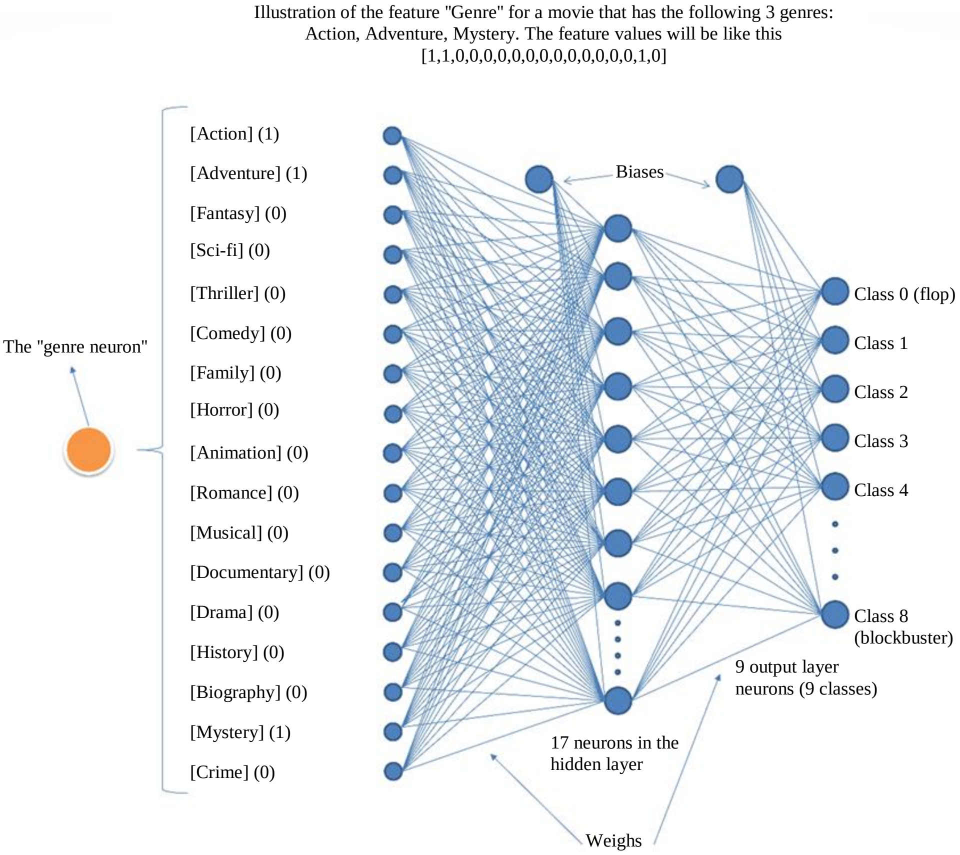

Fig. 13: Illustration of the feature “Genre”

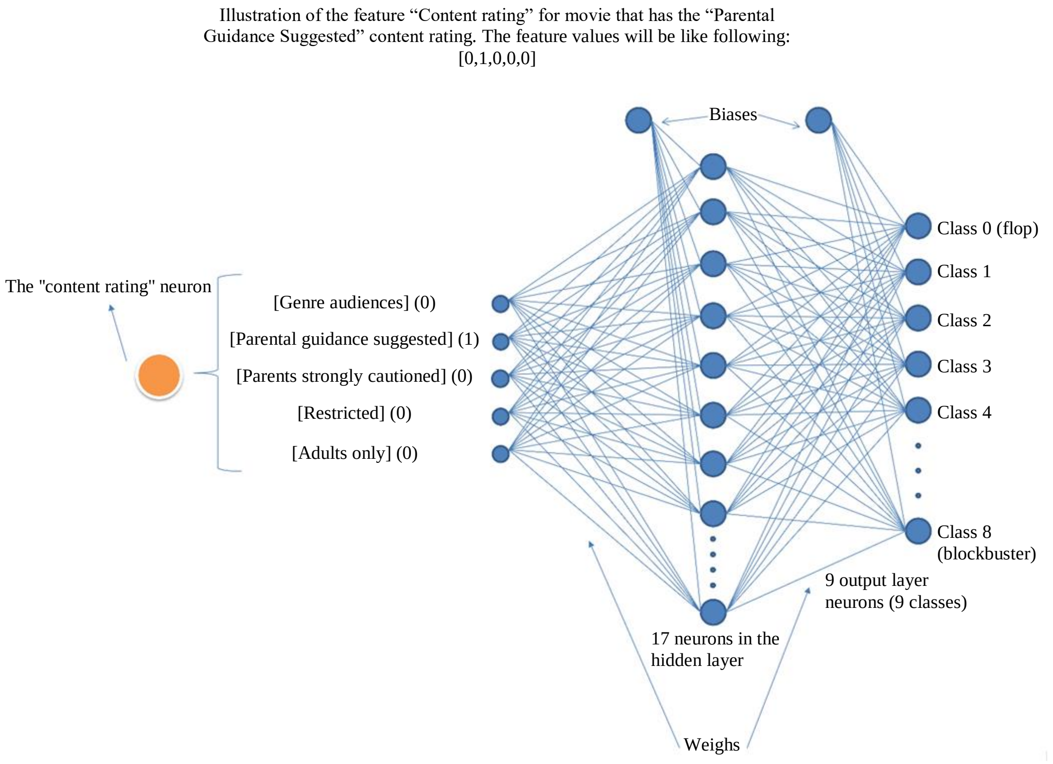
Illustration of the feature “Content rating” for movie that has the “Parental Guidance Suggested” content rating. The feature values will be like following: [0,1,0,0,0]
Biases
The "content rating" neuron
[Genre audiences] (0)
[Parental guidance suggested] (1)
[Parents strongly cautioned] (0)
[Restricted] (0)
[Adults only] (0)
Class 0 (flop)
Class 1
Class 2
Class 3
Class 4
Class 8 (blockbuster)
9 output layer neurons (9 classes)
17 neurons in the hidden layer
Weighs

Fig. 14: Illustration of the feature “Content rating’’

That’s why experiments are also carried out with logistics or sigmoid activation function which has the following form:

$$f(x)=\frac{1}{1+e^{-x}} \quad (8)$$

It generates output between 0 and 1. Hence it is more suited whenever we want the probabilistic distribution of the classes. A better sigmoid activation function is the tanh activation function which gives a broader range between -1 and 1 to work with. It has the following form:

$$f(x)=\frac{1-e^{-2x}}{1+e^{-2x}} \quad (9)$$

Most used activation function nowadays is the relu activation function which converts any input that is negative to 0. It has the following form:

$$f(x)=\max(0,x) \quad (10)$$

One of the most important step in implementing a MLP model is identifying the number of hidden layers and the number of neurons at each of those layers. As there is no direct rule to find out the actual numbers so we experimented with a few configurations. A rule of thumb as explained by Marsland (2014) is to use half of the number of neurons of the input layer. Figure 15 represents our findings. Our input layer contains 35 neurons. That means at around 17 or 18 neurons should give us the best accuracy in the hidden layer. It can be observed from Fig. 15 that the highest accuracy is achieved when we use 17 neurons at the hidden layer which is almost equal to half of the number of neurons at the input layer. Here both bingo and one class away accuracy are considered. So, a bingo accuracy of 56.4% is achieved using the MLP model. The relu activation function is used to conduct this experiment because it is the most widely used activation function. However, after conducting this experiment we also carried out experiments using identity, logistic and tanh activation functions. Figure 16 represents the comparison among different activation functions. It can be observed that the highest one class away accuracy (85.31%) is achieved by using the relu activation function and it also achieves fairly good bingo accuracy of 56.46%.

Finally, experiments are carried out to find out the number of hidden layers required to get the best performance out of our MLP model. Figure 17 shows that one hidden layer produces the best bingo and one class away accuracy.

So, after all these tunings final configuration of the MLP model that gave us the best accuracy is represented in Table 2.

Table 2: Neural network configuration

Number of hidden layers	1
Number of Input layer neurons	35
Number of hidden layer neurons	17
Number of output layer neurons	9
Initialization of weights	Between $-\frac{1}{\sqrt{n}}$ and $\frac{1}{\sqrt{n}}$ **;** where *n* represents number of features (Marsland, 2014)
Activation function	relu
Maximum iterations	1400

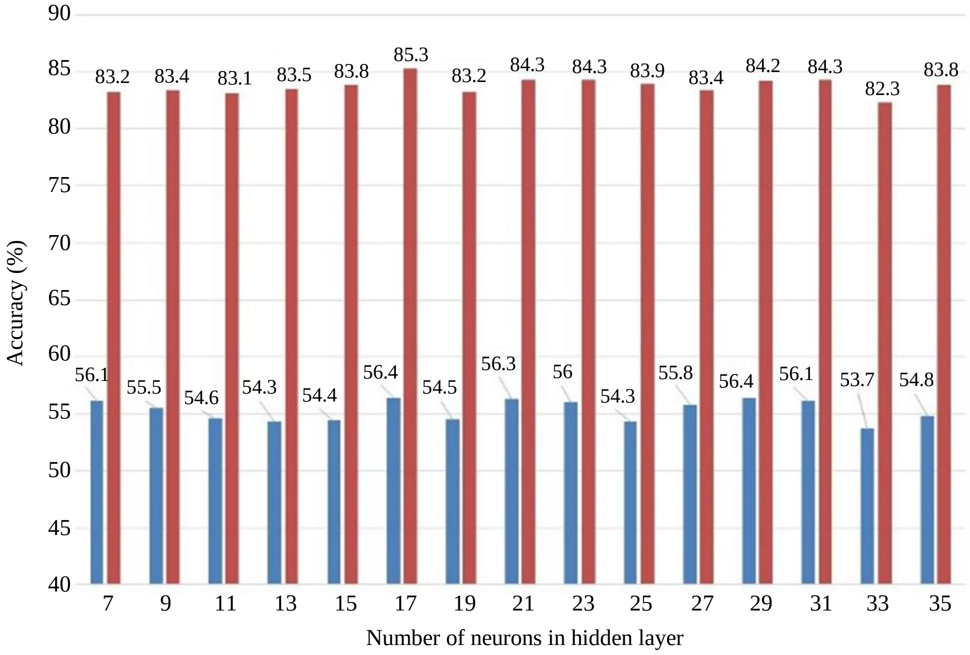

Fig. 15: Experimenting with different number of neurons in hidden layer (bingo accuracy)

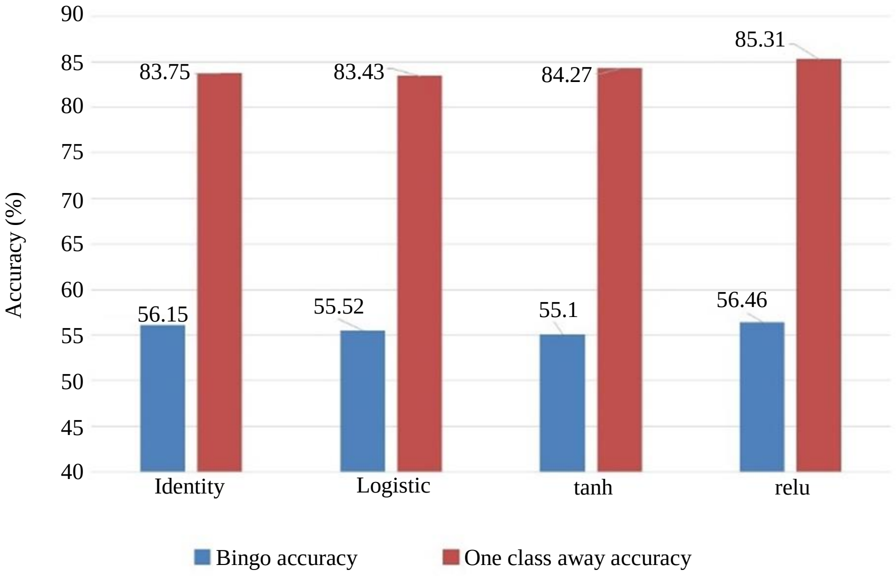

Fig. 16: Experimenting with different activation functions

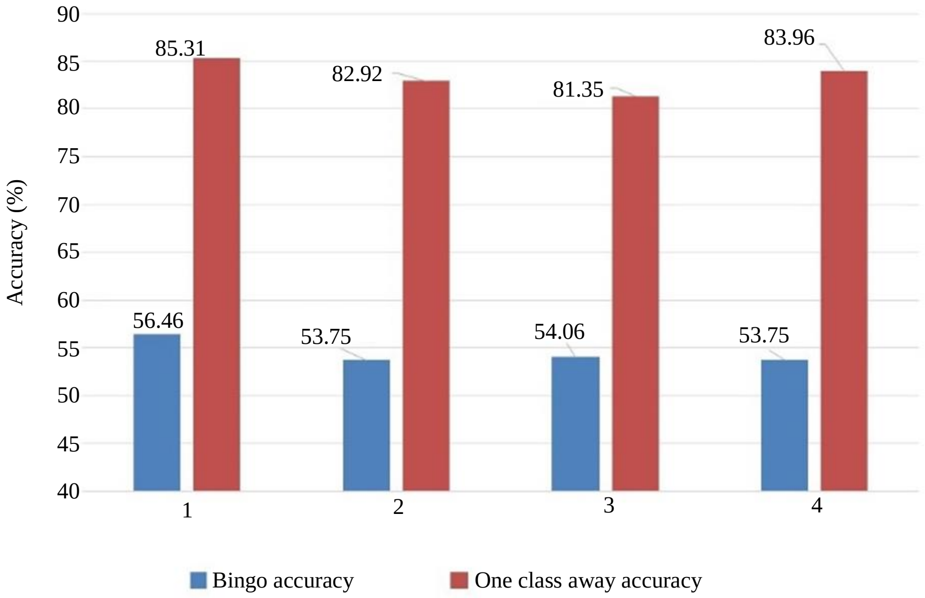

Fig. 17: Experimenting with different hidden layer size

Comparing the Best Machine Learning Approaches

Figure 18 shows the comparison of different machine learning approaches that achieved the best performance. It represents both bingo and one class away accuracy. From the comparison, we see that MLP model performed the best. It achieves better bingo and one class away accuracy than other models.

Comparing with Other Architectures

After comparing the machine learning models, it can be seen that the best performance is achieved by using the MLP model. So, this MLP model is compared with some other architectures from the literature. In all the experiments the dataset that is used is the "IMDB 5000 Movie Dataset" (2020). Abel *et al*. (2010) used Naïve Bayes and SVM. They reported that SVM achieved better accuracy (51.88% bingo accuracy, 80.42% one class away accuracy). They focused on the number of appearances of the movie title in social blogs to predict the outcome of the film. Kim *et al*. (2015) used two different architectures Support Vector Regression (SVR) and KNN to predict movie revenue. SVR generates continuous values while predicting class labels. So, the nearest class is taken with the help of rounding. For example, if it generates 7.34 then it is considered that the movie will generate profit within the range of class 7 and, if it generates 3.6 then it can be said that the movie will generate profit within the range defined by class 4. As can be seen from Fig. 19 the bingo accuracy of the SVR model and the KNN model is 52.19% and 53.02% respectively and the one class away accuracy of the SVR and the KNN model is 83.85% and 82.08% respectively. Lash and Zhao (2016) used the features: actors, directors and the casts, genre and MPAA ratings and the release time and achieved the best performance using the Logistic Regression model. Figure 19 shows that the architecture of Lash and Zhao (2016) achieved 54.89% bingo accuracy and 82.92% one class away accuracy. Quader *et al*. (2017) used SVM and MLP model to predict movie revenues. They claimed that their MLP model performed the best. Along with MLP, they used 15 features to predict the revenue generated by the film. They focused on IMDb rating, MPAA or content rating, User Review, Critics Review, IMDb votes, Budget and Director's popularity as their features. However, the popularity of actor/actress, genre, sequel, language of the film are not considered in their model. From Fig. 19 we can see that the bingo accuracy (56.6%) of their model is almost similar to our model. But when one class away accuracy (80.43%) is considered then our model performs better. Xiao *et al*. (2017) designed their Linear Regression model primarily based on the popularity of cast which includes the director, producer, actor/ actress, supporting actor/actress and the genre of the film. Figure 19 shows that their model achieves a bingo accuracy of 42.91% and the one class away accuracy is 81.56%. Walanaraya *et al*. (2018) used three different regression algorithms: Linear, Polynomial and SVR. Linear Regression was reported to have the best performance in their research. It can be observed from Fig. 19 that the bingo accuracy of their model is 42.19% and one class away accuracy of their model is 78.95%. Like SVR, Linear Regression also produces continuous-valued output. So, the nearest class label is chosen with the help of rounding. Galvão and Henriques (2018) used regression, decision tree and neural network to predict movie revenue. However, they achieved the best accuracy by using a MLP model which consists of 3 hidden layer neurons. They focused on the following features: Sequel, MPAA rating, Genre, Budget, Oscars, Awards, Directors, Actors, Season, Spectators and Critics. Factors like the popularity of the movie in the social networking sites and the IMDb score before the release of the film, the duration and language of the film were not considered in their research. Figure 19 shows that the bingo and one class away accuracy of their model is 55.42% and 83.33% respectively.

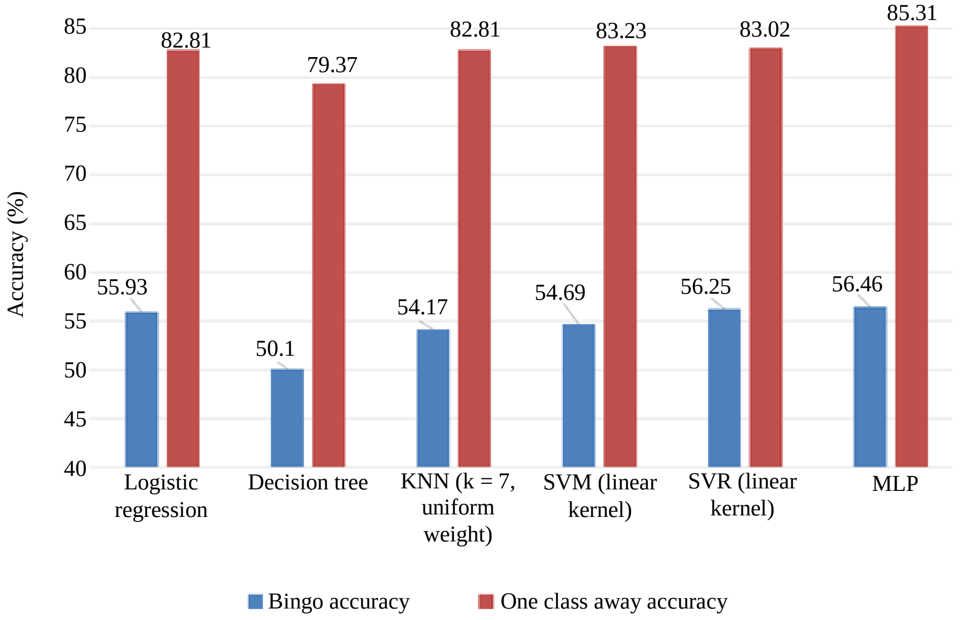

Fig. 18: Comparison among different machine learning algorithms

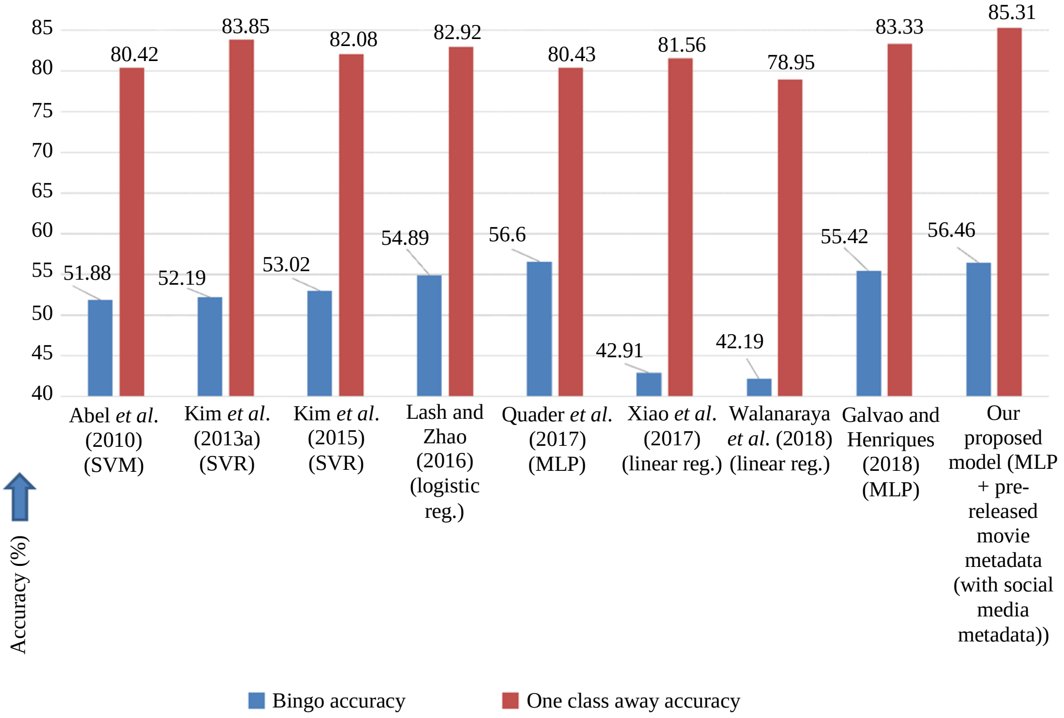

Fig. 19: Comparison with existing methodologies

It can be observed from the above analysis that our proposed model performed fairly well in comparison with other existing methodologies as it achieved an impressive one class away accuracy of 85.31% and if bingo accuracy is considered then it has 56.46% accuracy. Also, it can be observed that both of the lowest-performing architectures used Linear Regression based models.

Conclusion

In our research, we tried to explore the problem of predicting revenue generated by a film by simply using the metadata of a film that is released earlier. No sales information is used while predicting the revenue as sales data is not available in the early weeks of the release of the film. We used machine learning models to predict revenues generated by films. The configurations of the models and an analysis of the performance of the different models are also provided. It can also be observed from the analysis that the MLP model performs better than other machine learning algorithms even though the dataset is not very large. We achieved a fairly good one class away accuracy of 85.31% using the MLP model along with our handcrafted features. We also tested our model on a larger dataset than other researches that are mentioned in this study. But this model can be further improved by using more sophisticated classification technologies like deep learning models. But for that we need a large dataset. So, our next step will be to test our model on a bigger dataset and to implement more sophisticated classification techniques to solve this problem of movie revenue prediction.

Acknowledgement

We would like to thank Prof. Dr Mohammad Shahidur Rahman, Dr. Ahsan Habib and Asif Mohammed Samir of Shahjalal University of Science and Technology for their valuable advices throughout this research.

Author’s Contributions

Quazi Ishtiaque Mahmud: Contributing to the conceptualization of the research, developing the methodology**,** analyzing, investigating the feature space, designing the machine learning models, data collection and preparation, data cleansing, running experiments, identifying the features, preparing the final manuscript.

Nuren Zabin Shuchi: Studying related works in the field, preparing the final manuscript, identifying the importance of individual features, contributing to the proofreading of the manuscript.

Fazle Mohammed Tawsif: Studying related works in the field, contributing to data collection and preparation and data cleansing.

Asif Mohaimen: Crawling data from various sources to run experiments, contributing the data collection and data cleansing.

Ayesha Tasnim: Supervising the research.

Ethics

It is testified by the authors that this article has not been published anywhere else and contains no ethical issues.

References

Abel, F., E. Diaz-Aviles, N. Henze, D. Krause and P. Siehndel, 2010. Analyzing the blogosphere for predicting the success of music and movie products. Proceedings of the International Conference on Advances in Social Networks Analysis and Mining, Aug. 9-11, IEEE Xplore Press, Odense, Denmark. DOI: 10.1109/ASONAM.2010.50

Ahmad, J., P. Duraisamy, A. Yousef and B. Buckles, 2017. Movie success prediction using data mining. Proceedings of the 8th International Conference on Computing, Communication and Networking Technologies, Jul. 3-5, IEEE Xplore Press, Delhi, India, pp: 1-4. DOI: 10.1109/ICCCNT.2017.8204173

Al-Batah, M.S., S. Mrayyen and M. Alzaqebah, 2018. Arabic sentiment classification using MLP network hybrid with naive bayes algorithm. J. Comput. Sci., 14: 1104-1114. DOI: 10.3844/jcssp.2018.1104.1114

Alsaffar, A. and N. Omar, 2015. Integrating a lexicon based approach and k nearest neighbour for Malay sentiment analysis. J. Comput. Sci., 11: 639-644. DOI: 10.3844/jcssp.2015.639.644

Asur, S. and B.A. Huberman, 2010. Predicting the future with social media. Proceedings of the International Conference on Web Intelligence and Intelligent Agent Technology, Aug. 31-Sept. 3, IEEE Xplore Press, Toronto, ON, Canada, pp: 492-499. DOI: 10.1109/WI-IAT.2010.63

Brewer, S.M., J.M. Kelley and J.J. Jozefowicz, 2009. A blueprint for success in the US film industry. Applied Econom., 41: 589-606. DOI: 10.1080/00036840601007351

Chang, B. and E. Ki, 2005. Devising a practical model for predicting theatrical movie success: Focusing on the experience good property. J. Media Econom., 18: 247-269. DOI: 10.1207/s15327736me1804_2

Chikersal, P., S. Poria and E. Cambria, 2015. SeNTU: Sentiment analysis of tweets by combining a rule-based classifier with supervised learning. Proceedings of the 9th International Workshop on Semantic Evaluation Proceedings of SemEval, Jun. 4-5, ACL, Denver, Colorado, pp: 647-651. DOI: 10.18653/v1/S15-2108

Choudhery, D. and C.K. Leung, 2017. Social media mining: Prediction of box office revenue. Proceedings of the 21th International Database Engineering and Applications Symposium, (EAS' 17), ACM, pp: 20-29. DOI: 10.1145/3105831.3105854

"Cornell Movie Review Dataset (polarity dataset v2.0)", 2004. http://www.cs.cornell.edu/people/pabo/movie-review-data/

Galvão, M. and R. Henriques, 2018. Forecasting movie box office profitability. J. Inform. Syst. Eng. Manage.

Gunter, B., 2018. Do sex, horror and violence sell movies? Predicting Movie Success Box Office.

IMDB 5000 Movie Dataset, 2020. https://data.world/data-society/imdb-5000-movie-dataset

Kim, D., E. Hwang and H. Choi, 2013a. A user opinion and metadata mining scheme for predicting box office performance of movies in the social network environment. New Rev. Hypermedia Multi., 19: 259-272. DOI: 10.1080/13614568.2013.835450

Kim, S.H., N. Park and S.H. Park, 2013b. Exploring the effects of online word of mouth and expert reviews on theatrical movies' box office success. J. Media Econom., 26: 98-114.
DOI: 10.1080/08997764.2013.785551

Kim, T., J. Hong and P. Kang, 2015. Box office forecasting using machine learning algorithms based on SNS data. Int. J. Forecast., 31: 364-390.
DOI: 10.1016/j.ijforecast.2014.05.006

Lash, M.T. and K. Zhao, 2016. Early predictions of movie success: The who, what and when of profitability. J. Manage. Inform. Syst., 33: 874-903. DOI: 10.1080/07421222.2016.1243969

Lee, K., J. Park, I. Kim and Y. Choi, 2018. Predicting movie success with machine learning techniques: Ways to improve accuracy. Inform. Syst. Frontiers, 20: 577-588. DOI: 10.1007/s10796-016-9689-z

Litman, B.R. and A. Kohl, 1989. Predicting financial success of motion pictures: The 80's experience. J. Media Econom., 2: 35-50.

Litman, B.R. and H. Ahn, 1998. Predicting Financial Success of Motion Pictures. In: The Motion Picture Mega-Industry, Litman, B.R. (Ed), Allyn and Bacon Publishing, Inc, Boston, MA.

Litman, B.R., 1983. Predicting success of theatrical movies: An empirical study. J. Popular Culture, 16: 159-175.

Marsland, S., 2014. Machine Learning: An Algorithmic Perspective. 2nd Edn., CRC Press, ISBN-10: 1466583339, pp: 457.

Mohamed, H., A. Atia and M.M. Mostafa, 2018. Sentiment analysis: Comparative study between GSVM and KNN. Am. J. Applied Sci., 15: 339-345. DOI: 10.3844/ajassp.2018.339.345

Mr. Go, 2013. https://www.imdb.com/title/tt2969458/

Neelamegham, R. and P. Chintagunta, 1999. A Bayesian model to forecast new product performance in domestic and international markets. Market. Sci., 18: 115-136.

Pennock, D.M., 2000. The real power of artificial markets. Science, 291: 987-988.
DOI: 10.1126/science.291.5506.987

People, 2020. https://people.com/

PSMLL, 2020. Python SKLearn Machine Learning Library.

Quader, N., M.O. Gani, D. Chaki and M.H. Ali, 2017. A machine learning approach to predict movie box-office success. Proceedings of the 20th International Conference of Computer and Information Technology, Dec. 22-24, IEEE Xplore Press, Dhaka, Bangladesh, pp: 1-7. DOI: 10.1109/ICCITECHN.2017.8281839

Ravid, S.A., 1999. Information, blockbusters and stars: A study of the film industry. J. Bus., 72: 463-92. DOI: 10.1086/209624

Rhee, T.G. and F. Zulkernine, 2016. Predicting movie box office profitability: A neural network approach. Proceedings of the 15th IEEE International Conference on Machine Learning and Applications, Dec. 18-20, IEEE Xplore Press, Anaheim, CA, USA, pp: 665-670. DOI: 10.1109/ICMLA.2016.0117.

Sachdev, S., A. Agrawal, S. Bhendarkar, B.R. Prasad and S. Agarwal, 2018. Movie box-office gross revenue estimation. Adv. Intell. Syst. Comput.

Sawhney, M.S. and J. Eliashberg, 1996. A parsimonious model for forecasting gross box-office revenues of motion pictures. Market. Sci., 15: 113-131.

Sharda, R. and D. Delen, 2006. Predicting box-office success of motion pictures with neural networks. Exp. Syst. Applic., 30: 243-254.
DOI: 10.1016/j.eswa.2005.07.018

Shim, S. and M. Pourhomayoun, 2017. Predicting movie market revenue using social media data. Proceedings of the International Conference on Information Reuse and Integration, Aug. 4-6, IEEE Xplore Press, San Diego, CA, USA, pp: 478-484. DOI: 10.1109/IRI.2017.68.

Simonoff, J.S. and I.R. Sparrow, 2000. Predicting movie grosses: Winners and losers, Blockbusters and sleepers. CHANCE, 13: 15-24.
DOI: 10.1080/09332480.2000.10542216

Sochay, S., 1994. Predicting the performance of motion pictures. J. Media Econom., 7: 1-20.
DOI: 10.1207/s15327736me0704_1

Treme, J., 2010. Effects of celebrity media exposure on box-office performance. J. Media Econom., 23: 5-16. DOI: 10.1080/08997761003590457

Walanaraya, P., W. Puengpipattrakul and D. Sutivong, 2018. Movie revenue prediction using regression and clustering. Proceedings of the 2nd International Conference on Engineering Innovation, Jul. 5-6, IEEE Xplore Press, Bangkok, Thailand, pp: 63-68. DOI: 10.1109/ICEI18.2018.8448610

Xiao, J., X. Li, S. Chen, X. Zhao and M. Xu, 2017. An inside look into the complexity of box-office revenue prediction in China. Int. J. Distributed Sensor Networks, 13: 155014771668484-155014771668484. DOI: 10.1177/1550147716684842

Zufryden, F.S., 1996. Linking advertising to box office performance of new film releases: A marketing planning model. J. Adv. Res.