Multimedia Tools and Applications (2024) 83:20975–20996
https://doi.org/10.1007/s11042-023-16319-4

Predicting movie success based on pre-released features

Zulfiqar Ali Memon[1] · **Syed Muneeb Hussain**[1]

Received: 25 February 2022 / Revised: 22 May 2023 / Accepted: 13 July 2023 /
Published online: 5 August 2023

Abstract
Movie making is a billion-dollar industry. Every month hundreds of movies get released and earn millions of dollars in revenue. However, majority of the movies fail to create an impact on the Box-Office and flop. This not only put a bad impression on the entire cast and crew but also creates a huge setback in financial terms. As a producer or investor, it is crucial for them to have some certainty that the money they are investing in will give a good return otherwise they'll lose all their capital eventually. The idea of this research is to predict based on certain pre-released variables of the movie, whether an upcoming movie is going to succeed or fail in monetary terms. Many researches have already been doing that in this domain based on different techniques and around different datasets. The novelty of this research is that the proposed approach is not only based on classical movie features, but incorporates all other dependencies as well such as star power, popularity of the cast, track record of director, and actors, to predict whether movie will succeed or fail and whether an investor should invest in the movie proposal or not. This article uses multiple machine learning algorithms and tested them over various evaluation metrics. Among them, CatBoostRegression and Stacking Regression outperformed the remaining by giving the maximum model accuracy of 83.84% and 83.5% respectively. The article have used IMDB Movies Extensive Dataset. This dataset contains information of movies from 1894 to 2020 and has at least 100 votes.

Keywords Box-Office · Revenue · Movie Success · Predictive Analytics · Machine Learning · Success Criteria

1 Introduction

Movies have become very important part of our lives, and these are considered significant medium for delivering specific message, science innovations, stories, history of some country, cultures or entertainment. Due to this much publicity and the importance of movies now days the knowledge and research about the film industry field are growing

✉ Zulfiqar Ali Memon
memon.zulfiqar@gmail.com

Syed Muneeb Hussain
muneebhussain94@gmail.com

[1] Department of Computer Science, National University of Computer and Emerging Sciences – (NUCES-FAST), Karachi, Pakistan

exponentially. Every year hundreds of movies are being released, some of them are low budget movies and some are very high budget movies depending upon the production and the scope of movie. Some of the released movies are Blockbusters of the year and some go flop or average rated movies, depending on the budget, gross collection and the user reviews of the movie. Almost all the high budget movies are produced with the help of multiple investors and producer of movie has to convince the investors to invest in the specific movie, and here comes the challenging task of convincing someone to invest money in such movie that can go flop or can be blockbuster and that is unpredictable at the stage of making decision either to invest in such movie project or not. The problem of predicting the success of any releasing movie has been widely pondered upon multiple times in the past by many researchers, there are many useful datasets are available through which it can be predicted the movie will be successful or not, but the problem we are focusing in this research study is that how one can predict the success or failure in terms of capital investment and the revenue it will generate.

2 Background

Although many research works have been done and are in progress to predict the success or failure of released movies using different techniques. For instance, some researches use user reviews to predict the ratings of the movie, some go for genres, some use movie features like the actors, directors, producers and some use sentiment analysis on user and critic reviews or social media platforms to predict the movie's success or failure. Most of the researches however predict the success of how the movie will perform, post its release or if it has already been released, bus this success is majorly based on popularity and ratings. Very few researches primarily focus on pre-release and pre-production features of the movie to predict the success of movie in terms of Return On Investment (ROI) and this is something that offers a vast research gap that still needs to be filled.

In this research study, we will consider pre-production and pre-released features of movies and make meta dataset from existing datasets this will be discussed in methodology section, and the prediction will be done by considering the movie features that either the movie is worth investing in or not, and also this will be predicted that which feature change in specific movie can make the movie successful. This will not only predict the successfulness of the movie but also will tell the revenue it will generate once it gets released.

Machine learning algorithms have got a huge success when we talk about predictive modeling for any study, the researches in the field of predicting the successfulness of any movie has also made use various machine learning techniques. And since machine learning algorithms improve managerial decision-making so researchers prefer to use these predictive models to analyze and predict the results. However, in the case of box-office gross of any specific movie problem, there are multiple factors/features to be considered, as the movies are not independent at all. Many graphs can be identified among various movies, for example, one movie can be connected to another movie or the story of other movie, if they share the producer, actors, release years, or even the genres. For instance, the reputation or director or the actor is important in the movie, i.e. if actor/director have already some hit movies, then there may be chances of movie to be good performing, and so on. Considering many of the important features, the proposed research study will focus on predicting the successfulness movie before the production.

The driving factor behind this research is that there are a lot of researches that are being carried out to predict the success of an upcoming movie in terms of rating, popularity factor etc. but very few focuses on the actual revenue prediction that would help the producers or investors to ensure that the money they are investing in will not go waste and the movie will do well at the box office or to some extent, have chances to do well at the box office.

3 Related work

Since this industry is a multi-billion dollar one therefore multiple research works have been carried out and many more are in progress. However, the prediction is done mostly based on the classical movie features.

The research presented in [13, 22] is based on predicting the box office collection of a movie compared to the actual collection rather than predicting the score or popularity rating of a movie. In this paper, the authors have used a dataset containing 21 features from Box Office Mojo of movies ranging from 1980 to 2018 and used an Ensemble machine learning algorithm to predict the box office of movies. Some of the features include movie title, daily gross, weekly gross, rank, budget, theatre, and gross overseas, etc. The prediction is done mostly based on the post-released movie features and does not include historical features like awards and accolades.

A comparative analysis of different classification algorithms to predict the success of a Hollywood movie before its release is presented in [16, 19]. The authors manually scraped the data from IMDb and selected the top 150 movies released each year from 2008 to 2017 in the United States. Some of the features in their dataset were: MPAA rating, genre, budget, gross USA, actors, actresses, director, and release date, etc. The authors have divided the movie development into three phases: pre-production, during production, and post-production. This research is based on pre-production and during production processes like movie editing, final touches, etc. And based on that they have predicted the successfulness of movies. The data source used is IMDB only, the prediction could have been better if the sources were heterogeneous. Although pre-production, this research also does not cater the prediction of profitability or recovery before the capital is invested.

The research in [8, 10] has also considered classical movie features like actors, directors, budget, etc. but has tried to assist the director of the movie before its release about its success or failure based on classification models. Another byproduct of their research is for the end-user, whether he/she should reserve the cinema show for that movie. Although, the authors have incorporated various data sources including IMDB and social media for popularity factor, this research still is based on pre-released movie features only and does not include pre-production features. So, the gap would still be that the capital has already been invested, it is just about the successfulness of that investment.

The authors of the research in [3, 4] is focused on predicting the IMDB ratings of movies and then use IMDB as a benchmark to predict the accuracy of the model. The authors have used movie features like actors, directors, screenplay, etc. to predict success. They have applied the SMOTE technique to balance out their dataset in the preprocessing phase and applied five different classification models and compared their results with respect to IMDB ratings. This research only uses the classical movie features from only one data source i.e. IMDB. Furthermore, this research does not focus on pre-production movie features and only gives the prediction once the capital has been invested movie is released.

The authors in [15, 17] are particularly focused on considering the number of awards/accolades earned by the major features of a movie and then used those accolades in contrast with other features to test the accuracy of their models in predicting successfulness of a movie. They have also checked the relationship of awards when combined with single, bi, and multi-featured variables. This research is limited to the Bollywood film industry only and only focuses on the post-production movie features only.

The research done by the authors in [12, 18] also considers only Bollywood movies and gives prediction only based on 5 classes ranging from being blockbuster to flop. This research also takes in classical movie features, but they have worked on creating a historical base of prominent movie features. The authors have compared the results of five different classifiers. This research is limited to the Bollywood film industry only and focuses on the post-production prediction of a movie.

The research done by authors in [7, 11] also uses IMDB as a benchmark to predict movie ratings. The authors have taken in 28 features in the dataset and firstly defined the correlation between all variables and then used different classifiers and compared their results in terms of accuracy. This research only focuses on the post-released prediction of IMDB ratings meaning the movie has already been released and whether it fails or succeed it won't help in saving the capital investment which could have been if the pre-production features were used. The sources were homogenous. They could have linked the existing dataset with other sources to complement the accuracies of their classifiers.

The study conducted by authors in [6, 9] combines the conventional attributes along with the social factors like reviews on social media platforms, YouTube hits and comments, and Wikipedia page edits, etc. They also performed sentiment analysis on the user reviews from Twitter and then mapped all of them to predict the success of a movie. Dataset is diversified but historical analysis is not done in this research, like creating a historical base for actors to achieve more accuracy. Furthermore, the research is based on pre-released features only and does not include pre-production features that could save capital investment if predicted correctly.

The research in [20, 21] inculcates post-related features along with pre-released features to predict the success of the movie. The authors have used different data extraction techniques to extract classical as well as social features to improve their accuracy. One of the highlights is that they have used star power to compliment the accuracies of their models. The dataset is relatively small and had only 755 movies. Furthermore, this research does not focus on pre-production movie features and only gives the prediction once the capital has been invested movie is released.

The research presented by authors in [2, 5] combines classical movie features and social factors to determine the success of a movie. It has taken classical features like actors, director, budget, etc. and has also extracted user reviews, ratings, sentiments from their comments, and critics' reviews to improve their accuracy. Historical analysis is not done in this research, like creating a historical base for actors, directors, or accolades earned by them which could have helped to achieve more accuracy. This research also focuses on pre-released features only and does not cater the prediction based on the pre-production features.

4 Methodology

The basic predictive modeling methodology involves the six steps as shown in Fig. 1. Many existing researches have been made in the areas of text mining, sentiment analysis, and predictive analytics to predict the success/failure of a movie. What makes this research

distinctive of other researches is that rather than only using features of a movie like actors, director, budget, genre, movie score, etc., we have worked extensively on creating the history of each feature. So just as an example, instead of just using the feature of actor, we have extended the dataset and see the relationship of those derived variables. As an example, we wanted to check the effect of count of actors and directors on the profitability of the movie. This helped us in building a knowledge base of the primary attributes like lead actor, lead director, writer and producer, and see if their track records such as number of awards and accolades have any impact on the prediction or not. This track record can be extended to many other variables as well such as box-office collection and budget of previous movies done by that actor, etc. In this way, we have made a portfolio of the primary attributes the helped us in predicting the success of a movie in terms of revenue.

The primary dataset that we have used is IMDB Movies Extensive Dataset. This dataset contains information of movies from 1894 to 2020 and has at least 100 votes. However, we have not used data for the movies prior to 1894 and used a window of recent 30 years of movies. After that we have used the features of those movies let's say count of actors, directors, producers, writers, primary actor, director, etc. Afterwards, we have included the data of awards and accolades etc. and then used those derived features to build up the predictive model.

The prediction will be done on world-wide gross revenue based on these features and would be helpful in predicting more accurately about the films that are yet to be produced.

4.1 Data preparation

The Primary Dataset that we have used is **IMDB Movies Extensive Dataset**, as this dataset contains information of movies ranging from 1894 to 2020 and have at least 100 votes. IMDb stores information related to more than 6 million titles (of which almost 500,000 are featured films).

There are five csv files in the primary dataset, which are given below:

- Movies.csv (85,855 movies with 22 attributes)
- Names.csv (297,705 cast members with 20 attributes)
- Ratings.csv (85,855 rating details with 49 attributes)
- Title_principals.csv (835,513 casting roles with 6 attributes)
- Awards.csv (1,885,525 records with 21 attributes)

Five datasets have been merged to form a single dataset for the proposed research problem. Other than the existing columns in the files, we have also made some derived columns, which we believe will help in refining the accuracy of the proposed model in future, there are: count_actors, actor_1_name, actor_2_name, actor_3_name, count_directors, director_name, count_producers, producer_name, count_writers, and writer_name. The

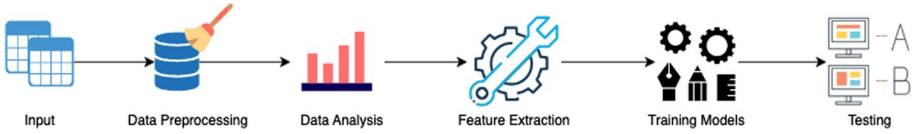


Fig. 1 Predictive Modeling Methodology

reason why we chose these extra variables is that we believe that having a good cast and good director will benefit a lot in terms of business. Therefore, we would like to see their effect also whilst training the proposed model.

There could be multiple actors, directors, writers, and producers. So, assuming that top 3 actors will help in achieving the target, we have taken top 3 actors from each movie (if there are multiple actors, otherwise the remaining actor fields would be left as blank) based on the **ordering** column in the **title_principals** dataset. Similarly, in case of multiple directors, writers, and producers, we have chosen the top one of each of them for every movie. Furthermore, we have used the awards datasets to derive another important attribute "**the star power",** which is the aggregated value of the achievements & accolades of these features and again would help in predicting with more accuracy. These attributes are: *actor_1_wins, actor_1_nominations, actor_2_wins, actor_2_nominations, actor_3_wins, actor_3_nominations, director_wins, director_nominations, producer_wins, producer_nominations, writer_wins, writer_nominations, film_wins, film_nominations.*

4.2 Data transformation and filtration

The target variable that we have predicted is **Box-office revenue** or the **Worldwide Gross Income**. Therefore, the first step was to convert the target variable and other important monetary variables such as **budget** and **usa_gross_income** into a uniform currency i.e., **US Dollar amount.** For this thing, we have used the currency_converter library of python.

The dataset used has movies from 1890s, but to avoid the biasedness, movies from the **year 1990 and onwards** have been taken for this study.

Secondly, this dataset has movies from many languages and regions. However, since we have restricted our research for Bollywood and Hollywood, therefore we have filtered out for only Hindi and English movies.

After filtering, ~ 28,163 rows, we were left in total with 59 columns. Not all columns were very useful, for e.g., most of the movies had only genre_1 and there were many nulls in genre_2 and genre_3 columns therefore we eliminated them. Similarly, the variables with count of null value greater than 80% were removed. After removal of these columns which were crossing the null threshold, some of the columns including budget, avg_votes, metascore were numerically imputed on the basis of year. The correlation between the columns were found with revenue and the columns with a correlation value between -0.2 to + 0.2 were removed. Ultimately, the final dataset had ~ 28,150 rows with 39 columns.

4.3 Exploratory data analysis

For exploratory data analysis, we have used **plotly** library of **Python** and **Microsoft Power BI.** The analysis is focused on the trends of budget and box-office revenue on different paradigms such as production companies, YoY growth/decline, star power and their impact, genre, and IMDB rating.

Figure 2 shows the split of production houses and their shares for the movies being released from 1990 to 2020.

Further (Fig. 3) is the pie chart for box-office revenue and budget utilized by the top production houses. The intent is to see the profitability of the production houses.

Looking at them, **Marvel Studios** seemed to produce only 12 movies in the last 30 years but has been the most successful among all with the revenue of greater than

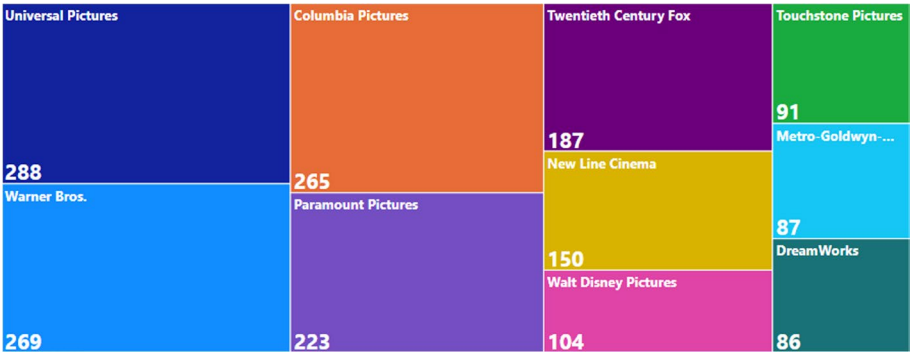


Fig. 2 Movies split of the top production companies

10 × of the budget. The total budget utilized is around $2.66bn (Fig. 3) but the worldwide gross revenue is $15.06 bn (Fig. 4).

Figure 5 is the genre-wise segregation of the movies released from 1990–2020. The genre **"Drama"** has been the most favorite since 1990 having the largest share of almost 26.3%, followed by Comedy, and Action genres and the finally, "Fantasy" having the lowest share with only 3.32% (Fig. 5).

Figure 6 tells us the YoY trend of growth and decline of production houses, their yearly total expenditure and the revenue earned from the movies. Looking at the year-on-year growth of each production houses, it is also evident that Marvel has performed better than the rest because it has an upward trend. It started in the end of 2010s and has been growing its revenue ever since. Recently, one of Marvel's movie "The Avengers: Endgame" has broken worldwide box office record for the highest grossing movie of all times. On the other hand, production houses like Warner Bros. and.

Universal Pictures have seen a decline since after 2010s. This may be related to the degraded performance of the movies or the better performance of other production houses like Marvel.

From the graph (Fig. 6), it can also be seen that from early 2000s to mid 2019s the profitability of movies has increased exponentially with respect to their budgets but we can see a drop at the end of 2019. This might be referred to as the seasonality effect due to COVID-19 because of which majority of the cinemas got shutdown and the films were

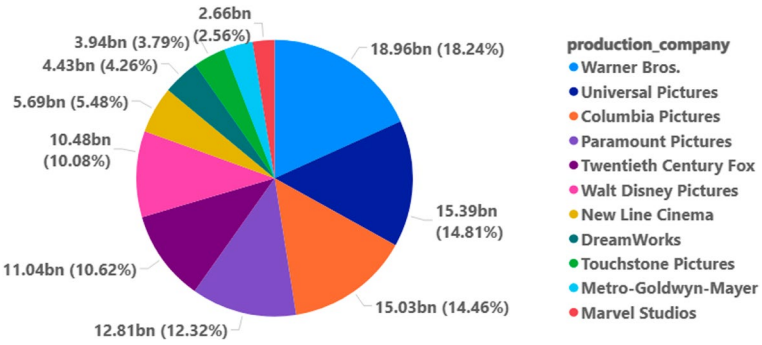


Fig. 3 Budgets utilized by top production houses

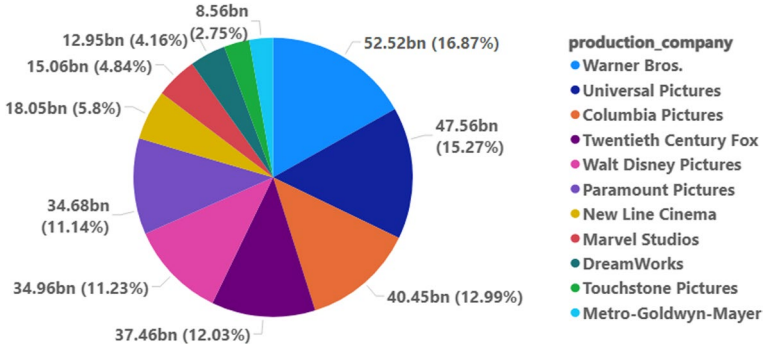


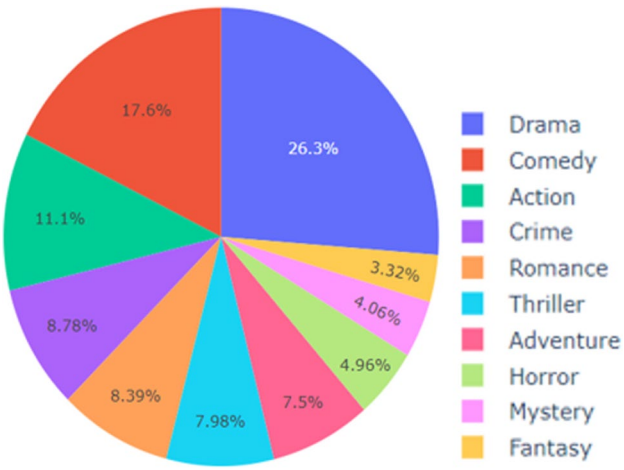
Fig. 4 Revenues generated by top production houses

released on online streaming services only thus affecting the release of movies in cinemas and therefore the capital gain from the cinema houses was lost.

Figure 7 is the year-wise distribution of sub of budgets and revenue generated by the movies released over the period of 30 years (Fig. 7). Again, there is an evident drop in the year 2020 due to COVID-19.

Next, we would like to see the relationship between the budget and box-office revenue and there seems to be a correlation between the budget and gross revenue i.e. the data-points got arranged in a linear manner when we sort both the budget and worldwide gross in ascending order meaning that the budget does have an impact on the revenue (Fig. 8). Similarly, we would like to do the same for revenue and their respective ratings. Again, it has linear relationship (Fig. 9). But, if we try and plot the reverse case that to check if the highly rated movies generate a substantial revenue compared to their budget incurred, then it becomes apparent that it is not true as we can see the highest rated movie, with an avg rating of 9.3, **The Shawshank Redemption** didn't earn a huge box-office revenue, merely $28 million with the budget of $25 million.

Fig. 5 Genre wise split



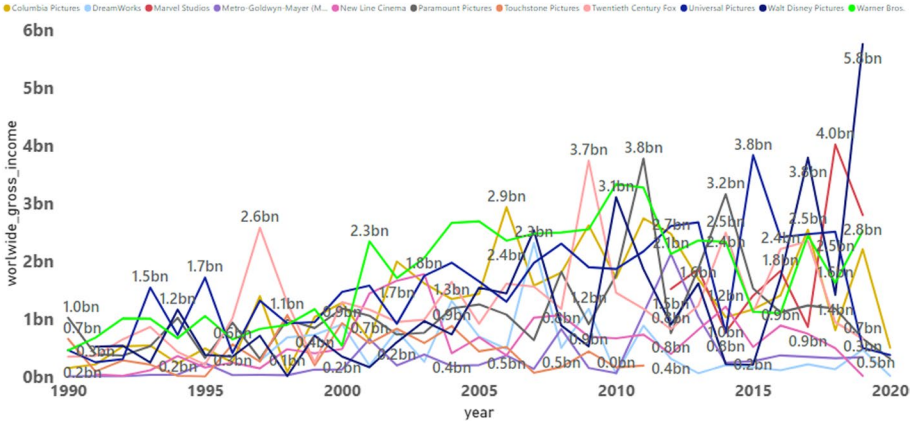


Fig. 6 YoY trend of revenue generated by production companies

Next, we need to understand the relationship between the box-office revenue and other features of the movie i.e. director, writer, and primary actor.

Following is a graph that shows the statistics for top 10 directors who have directed commercially successful films and their impact on the films they direct. As we can see in Fig. 10. the director **James Cameron** (the director of Titanic and Avatar which were both record breaking successful movies) tops the list with the least budget utilization and maximum gross revenue generation and that too with only 4 films since 1990 to 2020 (Fig. 10).

Similar to the aforementioned graph, Fig. 11. shows the impact of writer's profile on the profitability of movies. As can be observed, the writer Christopher Marcus seems to be the most impact creating writers as he has written only 9 movie's script but those movies have generated more than $8bn (Fig. 11).

Similar to the aforementioned graph, the Fig. 12. shows the relationship of primary actor in a movie and their star impact on the profitability of that movie. We have selected only top 10 actors whose castings have generated the maximum revenue. **Robert Downey Jr.** tops the list, this is due to his affiliation with the Marvel franchise. This is the reason why all the four things, that is, they Marvel Studios, the director Anthony Russo, the writer Christopher Marcus, and the actor Robert Downey Jr. have seen to be the most revenue generating factors among their respective lists for the Marvel (Fig. 12).

Hence, it can be concluded that the presence of a famous personality in any movie plays an important role and this does not depend upon the number of movies that person was associated with, but the performance of that person with the minimum amount of movies. In the examples, shown before, the persons James Cameron, Christopher Marchus,

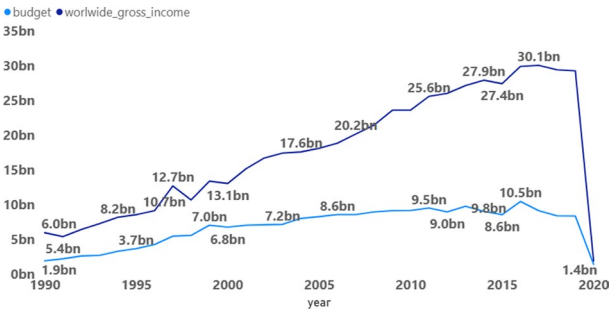


Fig. 7 Budget and Revenue comparison

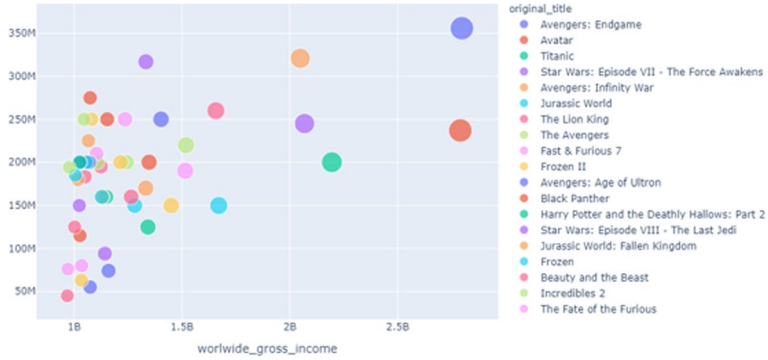


Fig. 8 Budget and revenue correlation

and Robert Downey Jr. had participated in the limited number of movies but those movie turned out to be super successful. Hence, there exists a strong correlation between these parameters and both the probable success or failure of the movie and the revenue margin.

4.4 Data encoding

The performance of the predictive model not only depends on the algorithm and hyperparameters but also on the nature of data it is taking as input. If the data is good, the results will be better and if the data is not in an appropriate form then it won't yield better results. Generally, the machine learning models take data in numerical form and as our dataset has continuous variables as well (like actor name, director name, producer name, etc.) therefore we have applied encoding techniques to convert these continuous features into numerical format so that the models can work appropriately.

1. *Label Encoding:*

The encoding technique to convert continuous features into numerical format that we have used in Label Encoding. This technique can be used if there is a sort of inherent

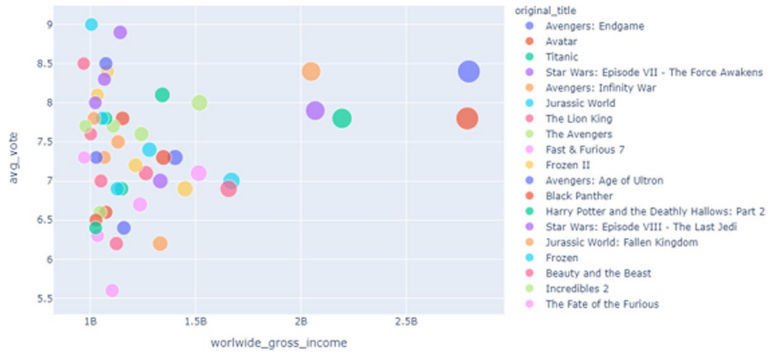


Fig. 9 Avg Rating and revenue correlation

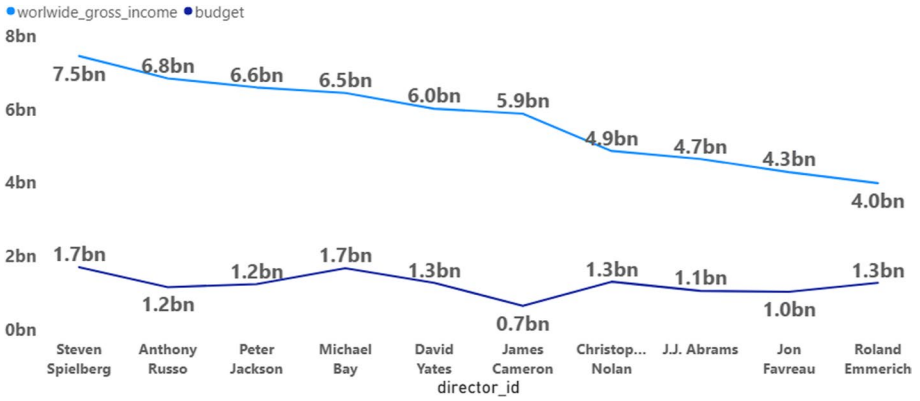


Fig. 10 Comparison of Budget and Revenue w.r.t. Director

category in the dataset. Since the continuous variables that we are dealing with like actor name is itself a category as we are interested in analyzing the effect of a particular actor in a movie therefore we will be interested if the encoder assigns the same integer value to that actor. This way the proposed model would be able to accurately predict the effect of casting the same actor in different movies and the difference in performance of both the movies on the box-office.

4.5 Evaluation metrics

Evaluation metrics let us check the performance of proposed model. Since in this experiment, we have used multiple algorithms and even made use of ensemble learning, therefore, we have used of the following metrics: R^2-Score, Mean Absolute Error, Mean Square/ Error, Explained Variance Score, Root Mean Square Error, and Normalized Mean Square Error.

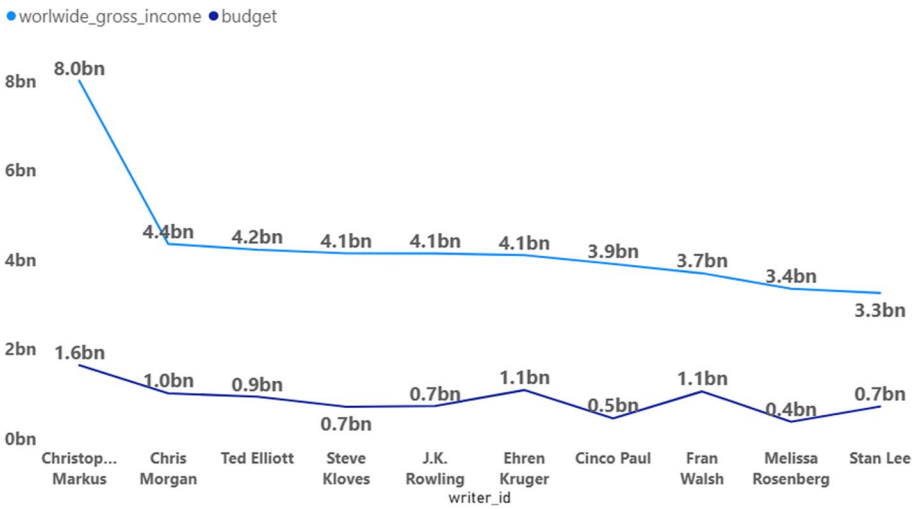


Fig. 11 Comparison of Budget and Revenue w.r.t. Writer

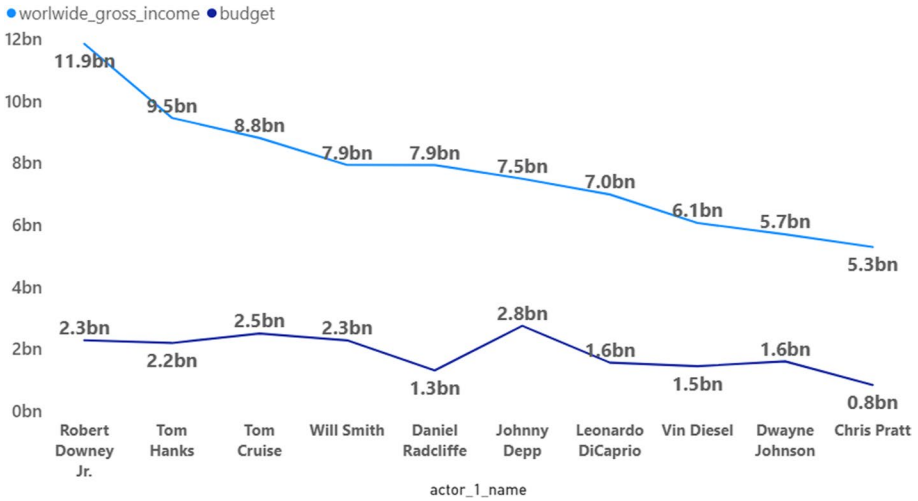


Fig. 12 Comparison of Budget and Revenue w.r.t. Actor

- R^2 Score:

$$R^2 = 1 - \frac{\text{sum squared regression (SSR)}}{\text{total sum of squares (SST)}},$$
$$= 1 - \frac{\sum \left(y_i - \hat{y}_i\right)^2}{\sum \left(y_i - \bar{y}\right)^2}.$$

- Mean Absolute Error:

$$\text{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} = \frac{\sum_{i=1}^{n} |e_i|}{n}.$$

- MSE:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} \left(Y_i - \hat{Y}_i\right)^2$$

- Explained Variance Score:

$$= 1 - \frac{Var(y - \hat{y})}{Var(y)}$$

- RMSE:

$$\text{RMSD} = \sqrt{\frac{\sum_{i=1}^{N} (x_i - \hat{x}_i)^2}{N}}$$

4.6 Experiment

For the experiment, we have used lineary regression to construct the baseline models and then also used some other algorithms as well. In the end, we have also used ensemble learning technique.

Here is the result we get for each of the respective techniques while 30% of the data is used for testing and 70% is used for training.

- Linear Regression:

Linear regression uses the training data points and infer the best regression line which has the least residual error. The residual error is the cumulative sum of change between the predicted value and the original value.

Linear regression is a supervised learning technique hence the data set consists of both the independent and output/dependent columns. To successfully predict the output variable, the model must be fit with the valid columns. The validity or effectiveness of the model depends on the column selection. The features that are in strong correlation with the output variables are picked while the irrelevant columns are discarded as they will ruin the model performance. One can simply find the correlation coefficient between the individual columns with the output variable and if the value tends to be closer with + 1 or -1, then that variable must be picked to fit the model. A correlation of 0 means that the output variable has least dependency over that input variable.

A linear regression line has an equation of the form $\boldsymbol{Y = a + bX}$, where **X** is the explanatory variable and **Y** is the dependent variable. The slope of the line is **b**, and **a** is the intercept (the value of **y** when $\mathbf{x} = 0$).

For our problem the algorithm produces these results while 30% of the data is used for testing and 70% is used for training:

Model Score % (R2 * 100) = 69.76%
R2 Score = 0.697569
Mean Absolute Error (MAE) = 2.497535e + 07
Mean Squared Error (MSE) = 2.181325e + 15
Explained Variance Score (EVS) = 0.697571
Root Mean Square Error (RMSE) = 4997.534699
Normalized MSE = 2960.568168
Max Error = 1.020606e + 09
Mean Absolute Percentage Error = 842.813880

- LGBM Regressor:

LightGBM extends the gradient boosting algorithm by improving it using an automatic feature selection technique as well as prioritizing on boosting examples with larger gradients. This can result in a dramatic speedup of training and improved predictive performance.

Light GBM is a high-performance, fast, distributed gradient boosting framework comprising a decision tree algorithm, used for classification, ranking and many other machine learning use cases.

As it is based on decision tree algorithms, it breaks the tree leaf wise with the best fit while other boosting algorithms split the tree in a depth wise or level wise manner. So, when growing on the same leaf in Light GBM, the leaf-wise algorithm can reduce more loss than the level-wise technique and hence produces much better accuracy. Also being very fast, it is termed as 'Light'.

For our problem the algorithm produces these results while 30% of the data is used for testing and 70% is used for training:

Model Score % (R2 * 100) = 82.06%
R2 Score = 0.820615
Mean Absolute Error (MAE) = 1.371347e + 07
Mean Squared Error (MSE) = 1.293843e + 15
Explained Variance Score (EVS) = 0.820651
Root Mean Square Error (RMSE) = 3703.169946
Normalized MSE = 1625.588863
Max Error = 9.552490e + 08
Mean Absolute Percentage Error = 404.410450

- XGB Regressor:

This algorithm goes by lots of different names such as gradient boosting, multiple additive regression trees, stochastic gradient boosting or gradient boosting machines.

XGBoost implements a gradient boosting algorithm that finds out the best alternative of all the models that are taking part in approximation. The improvements include computing second order gradients that have reduced the time and computation to coverage to the best. It has regularized the terms which improve model generalization. XGBoost can be used directly for regression predictive modeling.

For our problem the algorithm produces these results while 30% of the data is used for testing and 70% is used for training:

Model Score % (R2 * 100) = 80.45%
R2 Score = 0.804488
Mean Absolute Error (MAE) = 1.542520e + 07
Mean Squared Error (MSE) = 1.410157e + 15
Explained Variance Score (EVS) = 0.804542
Root Mean Square Error (RMSE) = 3927.492575
Normalized MSE = 1828.496672
Max Error = 9.487456e + 08
Mean Absolute Percentage Error = 452.22248

- CatBoost Regressor:

CatBoost originated from two words "Category" and "Boosting". CatBoost is a robust machine learning algorithm that can be used for a variety of business use cases. It can be used with diverse data points to provide the best in class prediction scores.

The salient features of CatBoost that make it dynamic include:

- It does not require a good and reasonable magnitude of data to produce SOTA results.
- It provides an out of the box solution for diverse data to fulfil the business need as best as possible.

For our problem the algorithm produces these results while 30% of the data is used for testing and 70% is used for training:

Model Score % (R2 * 100) = 83.84%
R2 Score = 0.838380
Mean Absolute Error (MAE) = 1.263704e + 07
Mean Squared Error (MSE) = 1.165705e + 15
Explained Variance Score (EVS) = 0.838415
Root Mean Square Error (RMSE) = 3554.861578
Normalized MSE = 1497.989668
Max Error = 9.091772e + 08
Mean Absolute Percentage Error = 398.232316

- GradientBoostingRegressor:

The idea of boosting came out of the idea of whether a weak learner can be modified to become better.

This algorithm goes by lots of different names such as gradient boosting, multiple additive regression trees, stochastic gradient boosting or gradient boosting machines.

Boosting is an ensemble technique where multiple models are attached to correct the errors made by existing models. Models are added sequentially until no further improvements can be done. A popular example is the AdaBoost algorithm that weights data points that are difficult to predict.

Gradient boosting is an approach where new models are created that predict the residuals or errors of previous models and then summed together to make the final prediction. It is called gradient boosting because it uses a gradient descent algorithm to reduce the loss when adding new models. This approach can be used for both regression and classification predictive modeling problems.

Boosting keeps changing the distribution of data points by taking the data points that were misclassified previously. It helps the model to learn which points can be reconsidered and which ones are already fine. It is performed iteratively in each round until the best performance has been achieved.

Gradient boosting involves three elements:

- A loss function to be optimized.
- A weak learner to make predictions.
- An additive model to add weak learners to minimize the loss function.

For our problem the algorithm produces these results while 30% of the data is used for testing and 70% is used for training:

Model Score % (R2 * 100) = 79.87%
R2 Score = 0.798731

Mean Absolute Error (MAE) = 1.543502e + 07
Mean Squared Error (MSE) = 1.451682e + 15
Explained Variance Score (EVS) = 0.798783
Root Mean Square Error (RMSE) = 3928.742567
Normalized MSE = 1829.660759
Max Error = 9.342819e + 08
Mean Absolute Percentage Error = 463.638092

- BayesianRidge:

Bayesian regression allows a natural technique to survive insufficient data or poorly distributed data by making linear regression using probability distributors rather than point estimates. The response 'y' is assumed to be drawn from a probability distribution rather than estimated as a single value.

Mathematically, to obtain a fully probabilistic model the response y is taken to be Gaussian distributed). One of the most useful types of Bayesian regression is Bayesian Ridge regression which estimates a probabilistic model of the regression problem.

For our problem the algorithm produces these results while 30% of the data is used for testing and 70% is used for training:

Model Score % (R2 * 100) = 69.76%
R2 Score = 0.697589
Mean Absolute Error (MAE) = 2.492875e + 07
Mean Squared Error (MSE) = 2.181184e + 15
Explained Variance Score (EVS) = 0.697591
Root Mean Square Error (RMSE) = 4992.870120
Normalized MSE = 2955.044101
Max Error = 1.020945e + 09
Mean Absolute Percentage Error = 849.537670

- AdaBoost Regressor:

It is a meta regressor. It first applies the model on the original data set and then assigns the weights to the last prediction based on residual errors. A value is assigned less weight if the last prediction has more errors.

For our problem the algorithm produces these results while 30% of the data is used for testing and 70% is used for training:

Model Score % (R2 * 100) = 77.30%
R2 Score = 0.773000
Mean Absolute Error (MAE) = 1.221830e + 08
Mean Squared Error (MSE) = 1.616877e + 16
Explained Variance Score (EVS) = 0.712979
Root Mean Square Error (RMSE) = 11,053.642962
Normalized MSE = 14,483.525692
Max Error = 8.158914e + 08
Mean Absolute Percentage Error = 5076.579669

- Huber Regressor:

Huber Regressor is a robust regression model that uses multiple approach to minimize the errors rather applying the conventional square loss function.

Mathematically, if the error is more, the penalty charged to the least-squares alternative is more while it is less for less error.

For our problem the algorithm produces these results while 30% of the data is used for testing and 70% is used for training:

Model Score % (R2 * 100) = 51.51%
R2 Score = 0.515104
Mean Absolute Error (MAE) = 2.111265e + 07
Mean Squared Error (MSE) = 3.497382e + 15
Explained Variance Score (EVS) = 0.527037
Root Mean Square Error (RMSE) = 4594.850625
Normalized MSE = 2502.685190
Max Error = 1.301203e + 09
Mean Absolute Percentage Error = 318.505591

- RANSAC Regressor:

RANSAC (RANdom SAmple Consensus) algorithm breaks down the data set into multiple inliners and applies the iterative approach from a subset of the data set. The subset is randomly picked by the system itself. Model is fitted again and again to produce better results.

For our problem the algorithm produces these results while 30% of the data is used for testing and 70% is used for training:

Model Score % (R2 * 100) = 78.00%
R2 Score = 0.780000
Mean Absolute Error (MAE) = 4.469621e + 07
Mean Squared Error (MSE) = 9.953674e + 15
Explained Variance Score (EVS) = 0.368138
Root Mean Square Error (RMSE) = 6685.522610
Normalized MSE = 5298.270812
Max Error = 3.323346e + 09
Mean Absolute Percentage Error = 885.328805

- SVM Regressor:

SVM is an approach that finds the best support vector to the discriminative line. It forms a plane that separates out the data. It facilitates us in setting the threshold for error that is acceptable. It uses a hyperplane in higher dimensions to fit the data.

The main goal is to reduce the coefficients rather than the errors. One can tune the hyperparameters (Epsilon) to gain the best possible accuracy. It does not work with whole data at a time, but chooses a few data points to complete the task.

For our problem the algorithm produces these results while 30% of the data is used for testing and 70% is used for training:

Model Score % (R2 * 100) = 4.96%
R2 Score = 0.049573
Mean Absolute Error (MAE) = 2.643377e + 07
Mean Squared Error (MSE) = 7.570203e + 15
Explained Variance Score (EVS) = 0.000005
Root Mean Square Error (RMSE) = 5141.378509
Normalized MSE = 3133.448669
Max Error = 1.659996e + 09
Mean Absolute Percentage Error = 409.816986

- Stacking Regression:

Stacked generalization is an ensemble technique that increases the performance of a model by combining multiple models together. It utilizes both the wrong and right prediction of models to improve the final prediction.

Two layers of models are involved, often referred to as level-0 models, and a meta-model that combines the predictions of the level-0 models. Meta Model is also termed as level-1 regressor.

- Level-0 Models (Base-Models): Models fit on the training data and whose predictions are combined.
- Level-1 Model (Meta-Model): Model that learns how to best combine the predictions of the base models.

The meta-model is trained on the predictions made by base models on out-of-sample data. The outputs from the base models provided as input to the meta-model may be real value in the case of regression, and probability values, or class labels in the case of classification.

In our case, we have used LGBMRegressor, XGBRegressor, CatBoostRegressor, and GradientBoostingRegressor as our base models while LinearRegression is used as the level-1 regressor.

For our problem the algorithm produces these results while 30% of the data is used for testing and 70% is used for training:

Model Score % (R2 * 100) = 83.5%
R2 Score = 0.835906
Mean Absolute Error (MAE) = 1.266114e + 07
Mean Squared Error (MSE) = 1.183552e + 15
Explained Variance Score (EVS) = 0.835977
Root Mean Square Error (RMSE) = 3558.249983
Normalized MSE = 1500.846721
Max Error = 8.959733e + 08
Mean Absolute Percentage Error = 368.523702

4.7 Results & validations

We have used eleven different algorithms and a combination of them for the ensemble learning to check if there is an added benefit of using ensemble technique on top of boosting algorithms. We have evaluated the mentioned algorithms using nine metrics to check

Table 1 Comparison of predictive models used for predicting movie revenue

Algorithm	Model Score	R^2 Score	Mean Absolute Error	Mean Squared Error	Explained Variance Score	Root Mean Square Error	Normalized Mean Square Error	Max Error	Mean Absolute Percentage Error
Linear Regression	69.76%	0.697569	2.497535e+07	2.1813e+15	0.697571	4997.535	2960.569	1.0206e+09	842.814
LGBM Regressor	82.06%	0.820615	1.371347e+07	1.2938e+15	0.820651	3703.170	1625.589	9.5524e+08	404.411
XGB Regressor	80.45%	0.804488	1.542520e+07	1.4101e+15	0.804542	3927.492	1828.450	9.4874e+08	452.223
CatBoost Regressor	83.84%	0.838380	1.263704e+07	1.1657e+15	0.838415	3554.862	1497.990	9.0917e+08	398.232
GradientBoosting Regressor	79.87%	0.798731	1.543502e+07	1.4516e+15	0.798783	3928.743	1829.661	9.3428e+08	463.638
Bayesian Ridge	69.76%	0.697589	2.492875e+07	2.1811e+15	0.697591	4992.870	2955.045	1.0209e+09	849.538
ADABoost Regressor	77.30%	0.773000	1.221830e+08	1.6168e+16	0.712979	11,053.643	14,483.526	8.1589e+08	5076.580
Huber Regressor	51.51%	0.515104	2.111265e+07	3.4973e+15	0.527037	4594.851	2502.686	1.3012e+09	318.506
RANSAC Regressor	78.00%	0.780000	4.469621e+07	9.9536e+15	0.368138	6685.523	5298.271	3.3233e+09	885.329
SVM Regressor	4.96%	0.049573	2.643377e+07	7.5702e+15	0.000005	5141.379	3133.449	1.6599e+09	409.817
Stacking Regression	83.5%	0.835906	1.266114e+07	1.1835e+15	0.835977	3558.250	1500.847	8.9597e+08	368.524

for model accuracy and for any inherent biasness within the dataset. For our specific problem, we have split the dataset into 70/30 split, 70% being our training set and 30% being our testing set. The evaluation is done on nine different metrics and the results are compiled in the Table 1.

5 Conclusion

Predicting movie success has always been critical since it affects not only the cast of the movie but the production house and producers as well. Many researches have been carried out in this domain involving various techniques of machine learning, some focus on classical features, some on social media approval, some on critical reviews, and many more. The effort done in this research work is heterogenous, we have not only used classical features but some derived variables as well like the impact of number of actors, producers, and directors, the awards and nominations earned by them and their relative effects. We have also found a correlation of having some specific parameters in the movie with its success such as the association of a particular actor with the movie does have an impact and is a useful parameter to predict the revenue.

To test the theory, we have used multiple machine learning algorithms and tested them over various evaluation metrics. Among them, CatBoostRegression and Stacking Regression outperformed the remaining by giving the maximum model accuracy of 83.84% and 83.5% respectively. Since CatBoostRegression is itself an ensemble model and gives off better accuracy than stacking regression therefore, it is safe to say that the inherent ensemble models work better and there is no need to stack up the individual models and manually create any other ensemble model.

6 Limitations

The limitation of this research is that it takes up only classical features and their historical and derived attributes like the actors count, awards earned by the lead actor, producer, director etc. But this research could be further extended if we incorporate social media factor as well just like it is done in [1, 2, 6, 8, 12, 14, 20]. Now the motivation behind this research was to only predict the revenue using regression technique to be able to tell accurately about the performance of the movie and to do that we have used only the classical features and their derivatives. But in future, if we incorporate the fan following for primary attributes or the YouTube likes or comments of the previous movies of those primary attributes, or if we could go further and combine it with the sentiment analysis to gauge responses of the critics/public of the previous movies of those primary attributes then it would surely result in improved performance. This particular thing is not in the scope of our research work but could be combined for future implementation.

Author's contributions All authors are Equally Contributed.

Data availability Data sharing not applicable to this article as no datasheets were generated or analyzed during the current study.

Declarations

Conflicts of interests/competing interests We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

References

1. Abbasi MA, Memon ZA, Durrani NM et al (2021) A multi-layer trust-based middleware framework for handling interoperability issues in heterogeneous IOTs. Cluster Comput 24:2133–2160. https://doi.org/10.1007/s10586-021-03243-1
2. Bhave A, Kulkarni H, Biramane V, Komsakar P (2015) Role of different factors in predicting movie success, in International Conference on Pervasive Computing
3. Bistri WR, Zaman Z, Sultana N (2019) Predicting IMDb rating of movies by machine learning techniques, in International Conference on Computing, Communication and Networking Technologies
4. Bosse T, Memon ZA, Treur J. Emergent storylines based on autonomous characters with mindreading capabilities, 2007 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'07), IEEE, pp. 207–214
5. Bosse T, Memon ZA, Treur J, Umair M (2009) An Adaptive Human-Aware Software Agent Supporting Attention-Demanding Tasks. In: Yang J-J, Yokoo M, Ito T, Jin Z, Scerri P (eds.), Proceedings of the 12th International Conference on Principles of Practice in Multi-Agent Systems, PRIMA'09. Lecture Notes in Artificial Intelligence, vol. 5925. Springer Verlag, pp. 292–307
6. Laeeq K, Memon ZA (2018) An integrated model to enhance virtual learning environments with current social networking perspective. Int J Emerg Technol Learn (Online) 13(9):252–268. https://www.academia.edu/download/73958405/5163.pdf
7. Samad F, Abbasi A, Memon ZA, Aziz A, Rahman A (2018) The Future of Internet: IPv6 Fulfilling the Routing Needs in Internet of Things. Int J Futur Gener Commun Netw. https://doi.org/10.14257/ijfgcn.2018.11.1.02
8. Bosse T, Memon ZA, Treur J (2008) Adaptive Estimation of Emotion Generation for an Ambient Agent Model. In: Aarts E et al. Ambient Intelligence. AmI 2008. Lecture Notes in Computer Science, vol 5355. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-89617-3_10
9. Hoogendoorn M, Klein MCA, Memon ZA, Treur J (2013) Formal Specification and Analysis of Intelligent Agents for Model-Based Medicine Usage Management. Comput Biol Med 43(5):444–457
10. Kashif UA, Memon ZA et al (2018) Architectural design of trusted platform for IaaS cloud computing. Int J Cloud Appl Comput (IJCAC) 8(2):47–65
11. Khan MA, Memon ZA, Khan S (2012) Highly Available Hadoop NameNode Architecture. In: 2012 International Conference on Advanced Computer Science Applications and Technologies (ACSAT), Kuala Lumpur, Malaysia, pp. 167–172. https://doi.org/10.1109/ACSAT.2012.52
12. Kashif UA, Memon ZA, Balouch AR, Chandio JA (2015) Distributed trust protocol for IaaS Cloud Computing. In: 12th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Islamabad, Pakistan, pp. 275–279. https://doi.org/10.1109/IBCAST.2015.7058516
13. Laghari A, Memon ZA, Ullah S, Hussain I (2018) Cyber Physical System for Stroke Detection. IEEE Access 6:37444–37453
14. Laghari A, Waheed-ur-Rehman, Memon ZA (2016) Biometric authentication technique using smartphone sensor. In: 13th International Bhurban Conference on Applied Sciences and Technology (IBCAST), Islamabad, Pakistan, pp. 381–384. https://doi.org/10.1109/IBCAST.2016.7429906
15. Bosse T, Hoogendoorn M, Memon ZA, Treur J, Umair M (2010) An Adaptive Model for Dynamics of Desiring and Feeling Based on Hebbian Learning. In: Yao Y, Sun R, Poggio T, Liu J, Zhong N, Huang J (eds) Brain Informatics. BI 2010. Lecture Notes in Computer Science, vol 6334. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-15314-3_3
16. Memon ZA, Treur J (2009) Modelling the Reciprocal Interaction between Believing and Feeling from a Neurological Perspective. In: Zhong N, Li K, Lu S, Chen L (eds) Brain Informatics. BI 2009. Lecture Notes in Computer Science, vol 5819. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-04954-5_12
17. Memon ZA, Treur J (2008) Cognitive and Biological Agent Models for Emotion Reading. In: Jain L, Gini M, Faltings BB, Terano T, Zhang C, Cercone N, Cao L (eds.), Proceedings of the 8th IEEE/

WIC/ACM International Conference on Intelligent Agent Technology, IAT'08. IEEE Computer Society Press, pp. 308–313
18. Memon ZA, Treur J (2010) On the Reciprocal Interaction Between Believing and Feeling: an Adaptive Agent Modelling Perspective. Cogn Neurodyn J 4(4):377–394
19. Memon ZA, Treur J (2012) An Agent Model for Cognitive and Affective Empathic Understanding of Other Agents. Trans Comput Collective Intell (TCCI) 6:56–83
20. Bosse T, Duell R, Memon ZA et al (2015) Agent-Based Modeling of Emotion Contagion in Groups. Cogn Comput 7:111–136. https://doi.org/10.1007/s12559-014-9277-9
21. Siddiqi S, Memon ZA (2016) Internet Addiction Impacts on Time Management That Results in Poor Academic Performance. In: 2016 International Conference on Frontiers of Information Technology (FIT), Islamabad, Pakistan, pp. 63–68. https://doi.org/10.1109/FIT.2016.020
22. Bosse T, Hoogendoorn M, Memon ZA, Treur J, Umair M (2012) A computational model for dynamics of desiring and feeling. Cogn Syst Res 19(20):39–61. https://doi.org/10.1016/j.cogsys.2012.04.002