A project report on

DECISION MODEL FOR PREDICTION OF MOVIE SUCCESS RATE

Submitted in partial fulfilment of the requirements for the degree of

Masters of Computer Applications

By

SHREYA GUPTA (19MCA0118)

CHETAN GARG (19MCA0092)

ROSHNI (19MCA0076)

SCHOOL OF INFORMATION TECHNOLOGY AND ENGINEERING (SITE)

JUNE 2020

Address: Vellore Institute of Technology, Vellore- 632014, Tamil Nadu, India.

CONTENTS

ABSTRACT

The purpose of this paper is to predict the success of any upcoming movie using Data Mining Tools. For this purpose, we have proposed a method which will analyze the cast and crew of the movie to find the success rate of the film using existing knowledge. Many factors like cast (actors, actresses, directors, producers), budget, worldwide gross, language will be considered for the algorithm to train and test the data. Two algorithms will be tested on our dataset and their accuracy will be checked.

INTRODUCTION

In this system we have developed a mathematical model for predicting the success class such as flop, hit, super hit of the movies. For doing this we have to develop a methodology in which the historical data of each component such as actor, actress, director, music that influences the success or failure of a movie is given is due to weight age and then based on multiple thresholds calculated on the basis of descriptive statistics of dataset of each component it is given class flop, hit, super hit label. Based on the weight age of historical data of each film crew the movie will be labeled as super hit, hit or flop.

This system helps to find out whether the movie is super hit, hit, flop on the basis of historical data of actor, actress, music director, writer, director, marketing budget and release date of the new movie. If the movie releases on weekend, new movie will get higher weight age or if the movie releases on week days new movie will get low weightage. The factors such as actor, actress, director, writer, music director and marketing budget historical data of each component are calculated and movie success is predicted. Due to this system, user can easily decide whether to book ticket in advance or not.

LITERATURE REVIEW

[15] They worked on an algorithm that would take into consideration the list of Actors, the popularity of said actors, the popularity of the director, the genre, the budget, the release year and gross. The files are collected from imdb.com in the form of x.list files which are then pre-processed and converted to their required form.

Using the gross-attribute as training element for the model. The dataset is converted into .csv files, after the pre-processing is done. They use logistic regression, followed by Gaussian Naïve Bayes Algorithm, decision trees, random forest classifiers, gradient boosting, artificial neural networks and support vector machines for the modelling.

Using the sentiment analysis vectors their Doc2vec algorithm provides high accuracy at sentiment polarity classification, allowing the producers to determine the necessary steps needed to make the movie successful.

[4] It worked on creating an algorithm that would analyze the blogosphere, i.e. the world of online blogs and use their tools to determine the success of a movie. The data set was gathered from October 1st to December 31st in 2008. Overall, the data set contained over 100 million blogposts. For each post Spinner provided the content and title of the post together with additional meta data, as ground truth they utilized Box Office Mojo's box office charts.

Using simple co-relation tests and simple pattern matching relevant blogs are identified, using the title of the movies, where it appears in the body, title or with a specific tf/idf score. Where they divided the movies into three classes of rise, stay and fall finding that support vector machine SMO can give up to 60% correct predictions.

[8] A mathematical model was developed to predict the success and failure of the upcoming movies based on several attributes. The data they gathered from movie databases was cleaned, integrated and transformed before the data mining techniques were applied. The processed data was analyzed on the basis of name, year of release, genre, directors, music directors and language. The mathematical model developed for to predict the success and failure would use various attributes using X^2 analysis like Genres vs Ratings or Actors vs Genres.

[9] Tried to approach the problem of movie predictions using Support Vector Machine and statistical reasoning trying to analyze public sentiments. This was done through gathering the data using IMDb and YouTube. Using 50000 movie reviews already created by Mass. they used feature, extraction techniques and polarity scores to create a list of successful or unsuccessful movies.

They created a feature matrix after getting the uni-gram and bi-gram in their dataset from the TF-IDF scores obtained by them and used positive and negative comments to help analyze their movie choices. Using IMDb ratings, they determine the successful movies from the unsuccessful ones.

[2] They tried to predict the box office performance of movies and showed that the features actors, the genre drama, sequel, among Content Ratings PG and R rated movies, the film's release period and budget and the number of first week screening all played an important and significant role in determining whether a film will be successful or not.

They used IMDb as the data source from where the sample data was drawn where upon co-relation analysis and multiple regression analysis was done on the data. According to their conclusion, brand power, actors or directors isn't strong enough to affect box office.

[3] They used neural networks to try and predict the box office performance of movies. The data sample they used was obtained from IMDb as the data source and they used the features MPAA rating, competition, star, genre, special effects, sequels and number of screens as analyzers of the obtained data sample.

Their neural network was able to obtain an accuracy of 36.9% and compromising mistakes made within one category an accuracy of a whopping 75.2%. They also were able to show that number of screens, high technical effects and high star value made more impact in deciding the significant role in deciding the fate of the film rather than competition, MPAA ratings or Genre.

[10] They used machine learning to try and create a decision support system for movie investment sector. Their study tried to predict approximate success rates for movies based on its profitability. This in turn would help investors from associated with risky ventures.

According to their assumption only profitable movies at the box office would be considered successes disregarding critic reviews and awards. They acquired their data from the data sources IMDb, Rotten Tomatoes, Metacritic and Box Office Mojo. Their data set consisted of 755 movies released between the years 2012 and 2015.

Initially having a dataset of 3183 movies, they removed movies whose budget could not be found or missed key features in the end a dataset of 755 movies was obtained. After Key feature extraction was completed. Using sentiment analysis, support vector machine and neural network analysis on these data sets it would give 48.41% exact match and 84.1% accuracy on predictions.

[11] Authors tried to approach this with the standard Regression Analysis and Support Vector Machine analysis. Their study tried to predict the success of a movie using regression and SVM. The data they collected was gathered from Box Office Mojo and Wikipedia. Their data was comprised of movies released in 2016.

They focused on the following features, Opening Date, Movie Name, Budget, Domestic Gross, International Gross, Total Gross, Trailer Views, Studio, Cast and Crew, Genre, Medium, Wikipedia Views and Rotten Tomato Score. The data was first pruned removing movies with low budgets and details not made public.

The data was then classified into three different categories: low budget, medium budget and high budget. With the classifications in place, the minimum expected gross of the movies was calculated. Then the distributed data was observed in real-time.

[5] An entirely different approach was followed. Instead of trying to predict the success of a movie. They tried to perform a network analysis on the teams that produce these movies and how it may lead to success.

The data was taken from the Internet Movie Database or IMDb as the data source, the data they obtained was from the years 1945 to 2017. For evaluating the movies, they used gross income and user rating provided in IMDb. Using the two variables they were able to co-relate a movie’s success with its public acceptance.

[12] They attempted to predict the success of movies using visualization of spatio-temporal data. According to them, twitter is a platform that can provide geographical as well as timely information, making it a perfect source for spatio-temporal models.

[13] The project is division of mathematical models that forecast to find the positive result of coming movies based on definite elements. The result is also in favor for the people who are interested in watching movie. The benefit of this model plays an important role in movie forecasting because it includes large contribution of investors. As, result cannot be forecast on the basis of a particular element. As per the mathematical model, it was achieved that one element is character which completes the achievement of the movie.

[14] In the project we use various designs for analyzing the results of Hollywood and Bollywood movies by testing sentiment information posted by the people as per their liking. The major work is to use of S-PLSA model. To achieve the accurate result, the first step is to study whole data and hundreds of documents that are cleared up, combined and remodeled ahead of that data mining approach. On the basis of P-N ratio we can divide the reviews into good, bad and worst and then threshold data to analysis the positive result i.e. blockbuster, average and below average.

[16] To analyze the movie success by concluding IMDb rating used in Factorization Machines algorithm for rating upcoming movies by collecting data from online reviews of the people. The structure was refined to collect data from various origin and develop a database having the 2017 movies released in the USA.

[6] We have established the efficiency for every analysis method as well as analyzed the dataset achieved from various types of data. The data achieved has been analyzed only after documents that are cleared up, combined and remodeled, which needs large amount of time required for this analysis. The possibility of more accurate result leads to a brilliant system efficient of creating proposal for a movie in pre-production, so that changing a precise director or actor, which will lead to amplification of the rating of the upcoming film.

[7] In the project, we per sued a Movie Investor Assurance System (MIAS) to compensate film contribution agreement at the early stage of film productions. MIAS determine from openly applicable ancient data derived from different origin and tries to analyze film progress depends upon the profitability. The examination focuses the influence of data analytics in framework intelligence system that backing business settlement.

[1] In the study, our investigation displays an agreement for more evolution in this area. Take more time to integrate more of the expert data applicable, and any use of essential language handling methods, more exotic arrangement in the data can convert apparent. Other scientific classifies is also experienced for executing opinions which leads to well-informed system.

[17] This work constructs an algorithm to find the positive results of forthcoming films depends on certain elements. But it doesn't depend on the Features similar to film. The number of people who are willing to watch the movie plays very important role for the positive results. As there is no use for the industry if no one is interesting to watch the movie. By seeing the number of tickets sold for seeing the movie indicates the number of people interesting in watching movie which plays very important role. Now, future work is to suggest some features for the positive results.

SURVEY TABLE

Sr. No.	Authors	Year	Methodology & Algorithms used	Application
1	M. Saraee, S. White & J. Eccleston	2004	Classification Analysis	some useful data mining on the IMDb data, and uncovered information that cannot be seen by browsing the regular web front-end to the database.
2	Chang, Byeng-Hee & Ki, Eyun-Jung	2005	They used IMDb as the data source from where the sample data was drawn where upon co-relation analysis and multiple regression analysis was done on the data.	According to their conclusion, brand power, actors or directors isn't strong enough to affect box office.
3	Sharda, Ramesh and Dursun Delen	2006	Used neural networks to try and predict the box office performance of movies.	Their neural network was able to obtain an accuracy of 36.9% and compromising mistakes made

				within one category an accuracy of a whopping 75.2%.
4	Abel, Fabian & Diaz-Aviles, Ernesto & Henze, Nicola & Krause, Daniel & Siehndel, Patrick	2010	Simple co-relation tests and simple pattern matching relevant blogs were found.	They divided the movies into three classes of rise, stay and fall finding that support vector machine SMO can give up to 60% correct predictions.
5	W. Viana, P. O. Santos, A. P. C. d. Silva and M. M. Moro	2014	They tried to perform a network analysis on the teams that produce these movies and how it may lead to success.	The data was taken from the Internet Movie Database or IMDb as the data source, the data they obtained was from the years 1945 to 2017.
6	Nikhil Chaudhari, Karthik Vardhrajan, Shashank Shekhar, Prabhjit Thind and Swarnalatha P	2016	Decision Tree Algorithm	A more accurate classifier is also well within the realm of possibility, and could even lead to an intelligent system capable of making suggestions for a movie in pre-production, such as a change to particular director or actor, which would be likely to increase the rating of the resulting film.
7	MICHAEL T. LASH AND KANG ZHAO	2016	Prediction Analytics, Network Analysis, Prescriptive Analytics	In this study, we proposed a movie investor assurance system (MIAS) to aid movie investment decisions at the early stage of movie productions.

				MIAS learns from freely available historical data derived from various sources, and tries to predict movie success based on profitability.
8	J. Ahmad, P. Duraisamy, A. Yousef and B. Buckles	2017	The mathematical model developed for to predict the success and failure would use various attributes using X2 analysis like Genres vs Ratings or Actors vs Genres	The data they gathered from movie databases was cleaned, integrated and transformed before the data mining techniques were applied.
9	Q. I. Mahmud, A. Mohaimen, M. S. Islam and Marium-E-Jannat	2017	Tried to approach the problem of movie predictions using Support Vector Machine and statistical reasoning trying to analyze public sentiments.	They used feature, extraction techniques and polarity scores to create a list of successful or unsuccessful movies. This was done through gathering the data using IMDb and YouTube.
10	N. Quader, M. O. Gani, D. Chaki and M. H. Ali	2017	Machine learning was used to try and create a decision support system for movie investment sector.	Initially having a dataset of 3183 movies, they removed movies whose budget could not be found or missed key features in the end a dataset of 755 movies was obtained. After Key feature extraction was completed.
11	V. Subramaniyaswamy , M. V. Vaibhav, R. V. Prasad and R. Yogesh	2017	Authors tried to approach this with the standard Regression Analysis and	The data they collected was gathered from Box Office Mojo and Wikipedia. Their

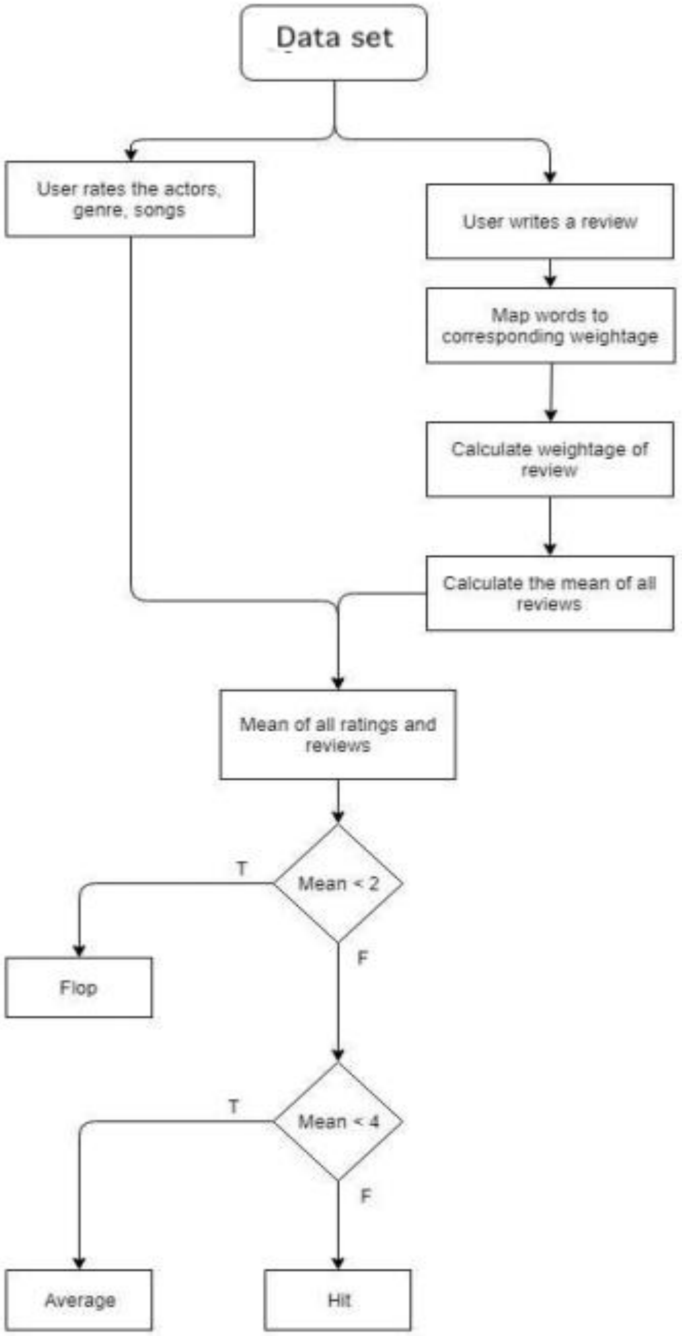
			Support Vector Machine analysis.	data was comprised of movies released in 2016.
12	A. Wijekoon, T. C. Sandanayake, A. Jayawardena, A. L. Y. Buddhini, U. K. D. G. S. Ariyawansha	2017	They attempted to predict the success of movies using visualization of spatio-temporal data.	According to them, twitter is a platform that can provide geographical as well as timely information, making it a perfect source for spatio-temporal models.
13	Javari a Ahmad, Prakash Duraisamy, Amr Yousef, Bill Buckles	2017	Mathematical model: Correlation Analysis	Mathematical Model is used to predict the success and failure of upcoming movies depending on certain criteria. Their work makes use of historical data in order to successfully predict the ratings of movies to be released.
14	Susmita S. Magdum, J.V. Megha	2017	Autoregressive Sentiment Model, S-PLSA Model (Sentiment Probabilistic Latent Semantic Analysis)	Using S-PLSA - the sentiment information from online reviews and tweets, we have used the ARSA model for predicting sales performance of movies using sentiment information and past box office performance.
15	Parikh, Yash & Palusa, Abhinivesh & Kasthuri, Shravankumar & Mehta, R & Rana, Dipti	2018	They used logistic regression, followed by Gaussian Naïve Bayes Algorithm, decision trees, random forest classifiers,	Using the gross-attribute as training element for the model. The data’s are converted into .csv files, after the

			gradient boosting, artificial neural networks and support vector machines for the modelling.	pre-processing is done.
16	Beyza Çizmeci, Sule Gündüz Ögüdücü	2018	Factorization Machine, Multivariate Linear Regression	Factorization Machines approach was used to predict movie success by predicting IMDb ratings for newly released movies by combining movie metadata with social media data.
17	Partha Chakraborty, Md. Zahidur Rahman, Saifur Rahman	2019	Clustering Model, Classification Algorithm	They developed a model to find the success of upcoming movies based on certain factors. The number of audience plays a vital role for a movie to become successful.

PROBLEM STATEMENT

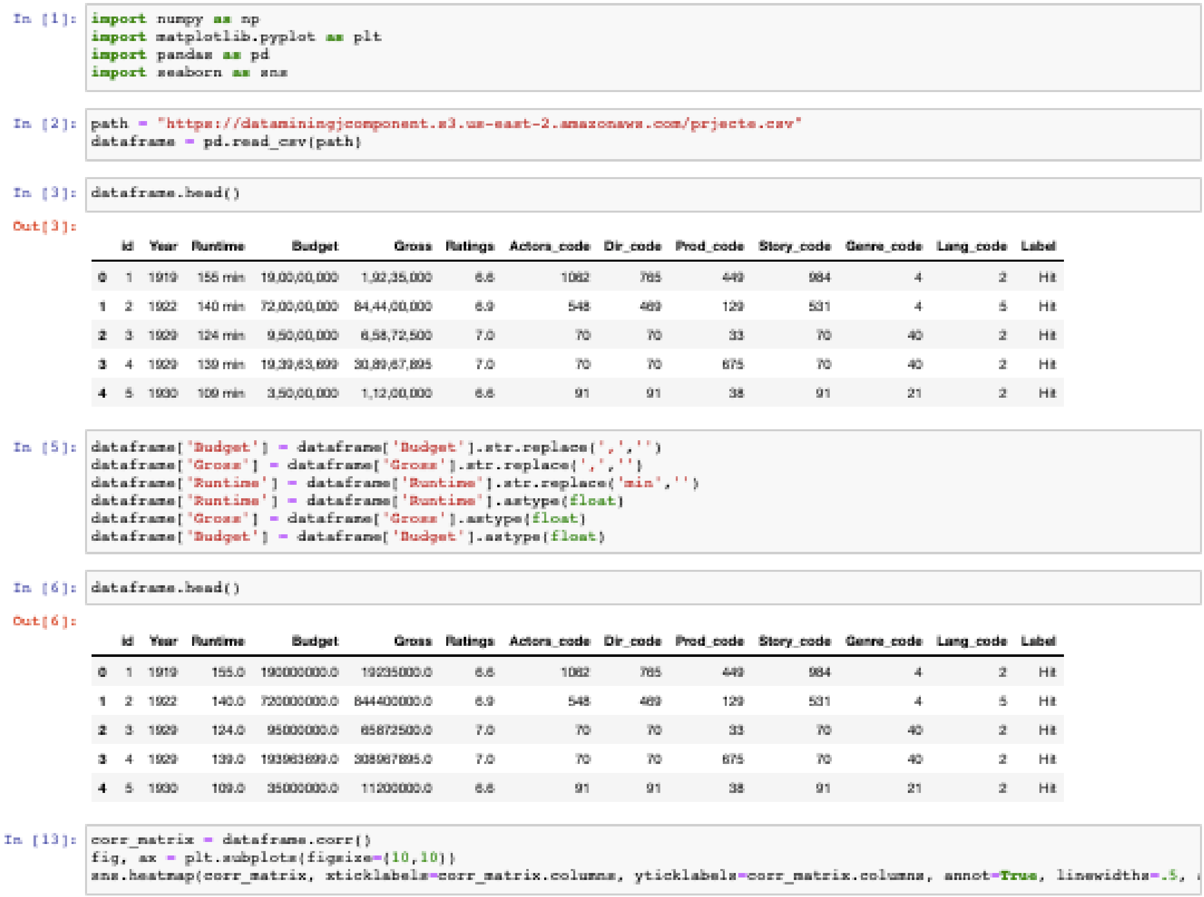
The method of using the ratings of the films the cast and crew has been is an innovative and an original way to solve the dilemma of film producers. Film producers have often trouble casting successful actors and directors and still trying to keep budget. Looking at the average ratings of each actor and director together of all the films they participated in should be able to give the producer a good idea on who to cast and who not to cast in a film that is to be out right now.

ARCHITECTURE DIAGRAM


Data set
User rates the actors, genre, songs
User writes a review
Map words to corresponding weightage
Calculate weightage of review
Calculate the mean of all reviews
Mean of all ratings and reviews
Mean < 2
T
F
Flop
Mean < 4
T
F
Average
Hit

IMPLEMENTATION

DATA PREPROCESSING & CORRELATION ANALYSIS



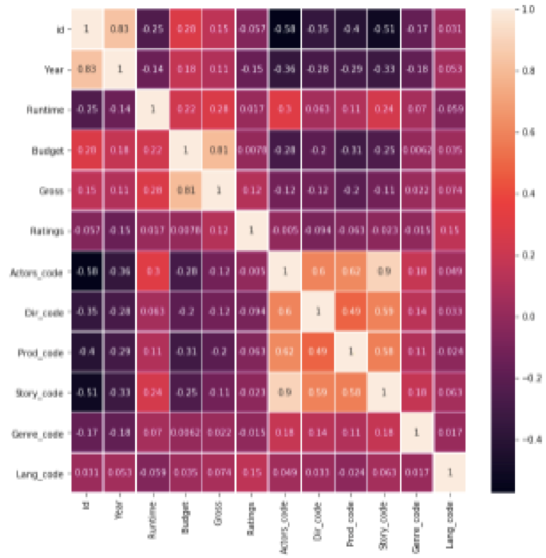
```
In [1]: import numpy as np
        import matplotlib.pyplot as plt
        import pandas as pd
        import seaborn as sns

In [2]: path = "https://dataminingjcomponent.s3.us-east-2.amazonaws.com/prjects.csv"
        dataframe = pd.read_csv(path)

In [3]: dataframe.head()

Out[3]:
```

	id	Year	Runtime	Budget	Gross	Ratings	Actors_code	Dir_code	Prod_code	Story_code	Genre_code	Lang_code	Label
0	1	1919	155 min	19,00,00,000	1,92,35,000	6.6	1062	765	449	984	4	2	Hit
1	2	1922	140 min	72,00,00,000	84,44,00,000	6.9	548	489	129	531	4	5	Hit
2	3	1929	124 min	9,50,00,000	6,58,72,500	7.0	70	70	33	70	40	2	Hit
3	4	1929	139 min	19,39,63,699	30,89,67,895	7.0	70	70	675	70	40	2	Hit
4	5	1930	109 min	3,50,00,000	1,12,00,000	6.6	91	91	38	91	21	2	Hit

```
In [5]: dataframe['Budget'] = dataframe['Budget'].str.replace(',','')
        dataframe['Gross'] = dataframe['Gross'].str.replace(',','')
        dataframe['Runtime'] = dataframe['Runtime'].str.replace('min','')
        dataframe['Runtime'] = dataframe['Runtime'].astype(float)
        dataframe['Gross'] = dataframe['Gross'].astype(float)
        dataframe['Budget'] = dataframe['Budget'].astype(float)

In [6]: dataframe.head()

Out[6]:
```

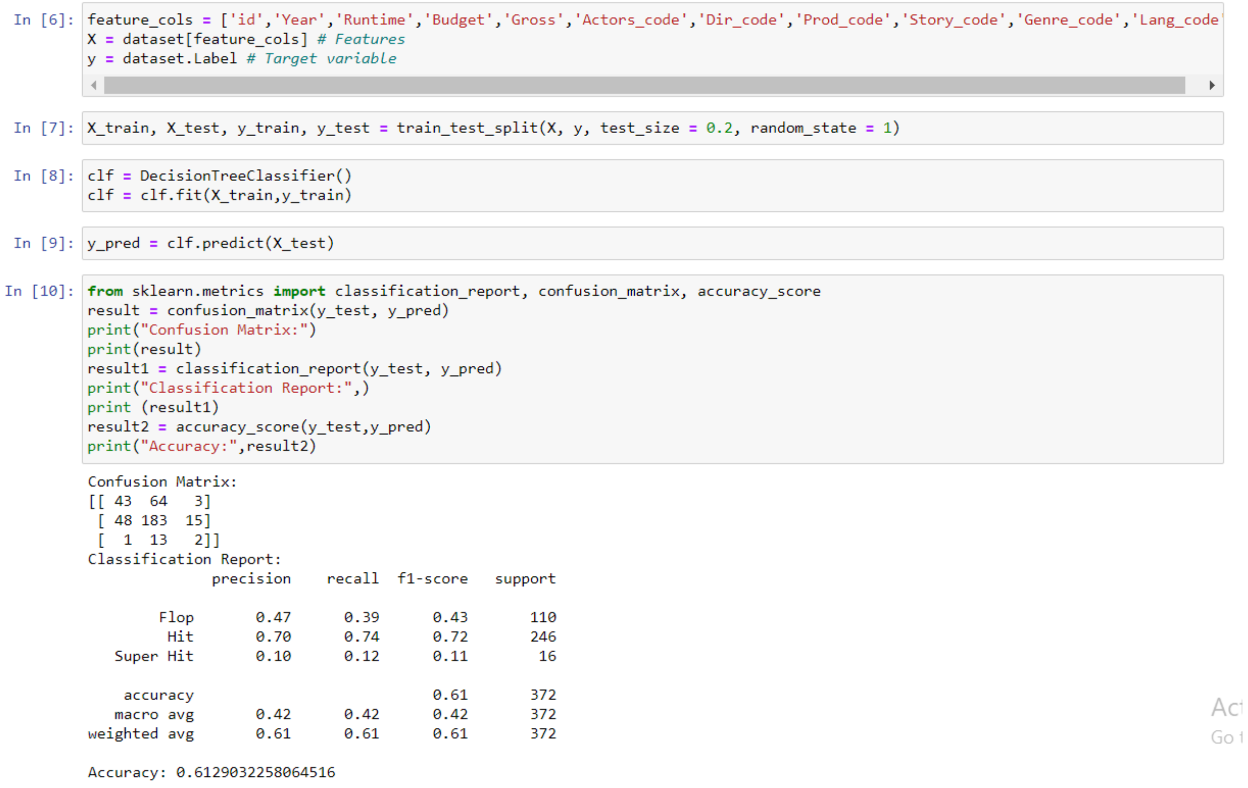
	id	Year	Runtime	Budget	Gross	Ratings	Actors_code	Dir_code	Prod_code	Story_code	Genre_code	Lang_code	Label
0	1	1919	155.0	190000000.0	19235000.0	6.6	1062	765	449	984	4	2	Hit
1	2	1922	140.0	720000000.0	844400000.0	6.9	548	489	129	531	4	5	Hit
2	3	1929	124.0	95000000.0	65872500.0	7.0	70	70	33	70	40	2	Hit
3	4	1929	139.0	193963699.0	308967895.0	7.0	70	70	675	70	40	2	Hit
4	5	1930	109.0	35000000.0	11200000.0	6.6	91	91	38	91	21	2	Hit

```
In [13]: corr_matrix = dataframe.corr()
         fig, ax = plt.subplots(figsize=(10,10))
         sns.heatmap(corr_matrix, xticklabels=corr_matrix.columns, yticklabels=corr_matrix.columns, annot=True, linewidths=.5, 
```

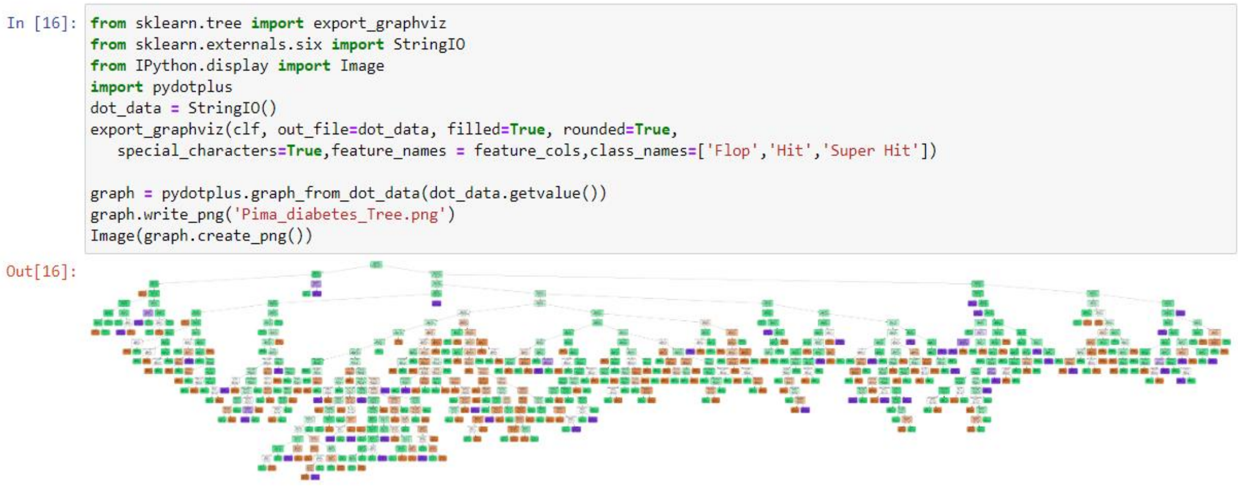


	id	Year	Runtime	Budget	Gross	Ratings	Actors_code	Dir_code	Prod_code	Story_code	Genre_code	Lang_code
id	1	0.83	-0.25	0.20	0.15	-0.057	-0.58	-0.35	-0.4	-0.51	-0.17	0.031
Year	0.83	1	-0.14	0.18	0.11	-0.15	-0.36	-0.28	-0.29	-0.33	-0.18	0.053
Runtime	-0.25	-0.14	1	0.22	0.28	0.017	0.3	0.063	0.11	0.24	0.07	-0.059
Budget	0.20	0.18	0.22	1	0.81	0.0078	-0.28	-0.2	-0.31	-0.25	0.0062	0.035
Gross	0.15	0.11	0.28	0.81	1	0.12	-0.12	-0.12	-0.2	-0.11	0.022	0.074
Ratings	-0.057	-0.15	0.017	0.0078	0.12	1	-0.005	-0.094	-0.063	-0.023	-0.015	0.15
Actors_code	-0.58	-0.36	0.3	-0.28	-0.12	-0.005	1	0.6	0.62	0.9	0.18	0.049
Dir_code	-0.35	-0.28	0.063	-0.2	-0.12	-0.094	0.6	1	0.49	0.59	0.14	0.033
Prod_code	-0.4	-0.29	0.11	-0.31	-0.2	-0.063	0.62	0.49	1	0.58	0.11	-0.024
Story_code	-0.51	-0.33	0.24	-0.25	-0.11	-0.023	0.9	0.59	0.58	1	0.18	0.063
Genre_code	-0.17	-0.18	0.07	0.0062	0.022	-0.015	0.18	0.14	0.11	0.18	1	0.017
Lang_code	0.031	0.053	-0.059	0.035	0.074	0.15	0.049	0.033	-0.024	0.063	0.017	1

APPLICATION OF DECISION TREE ALGORITHM

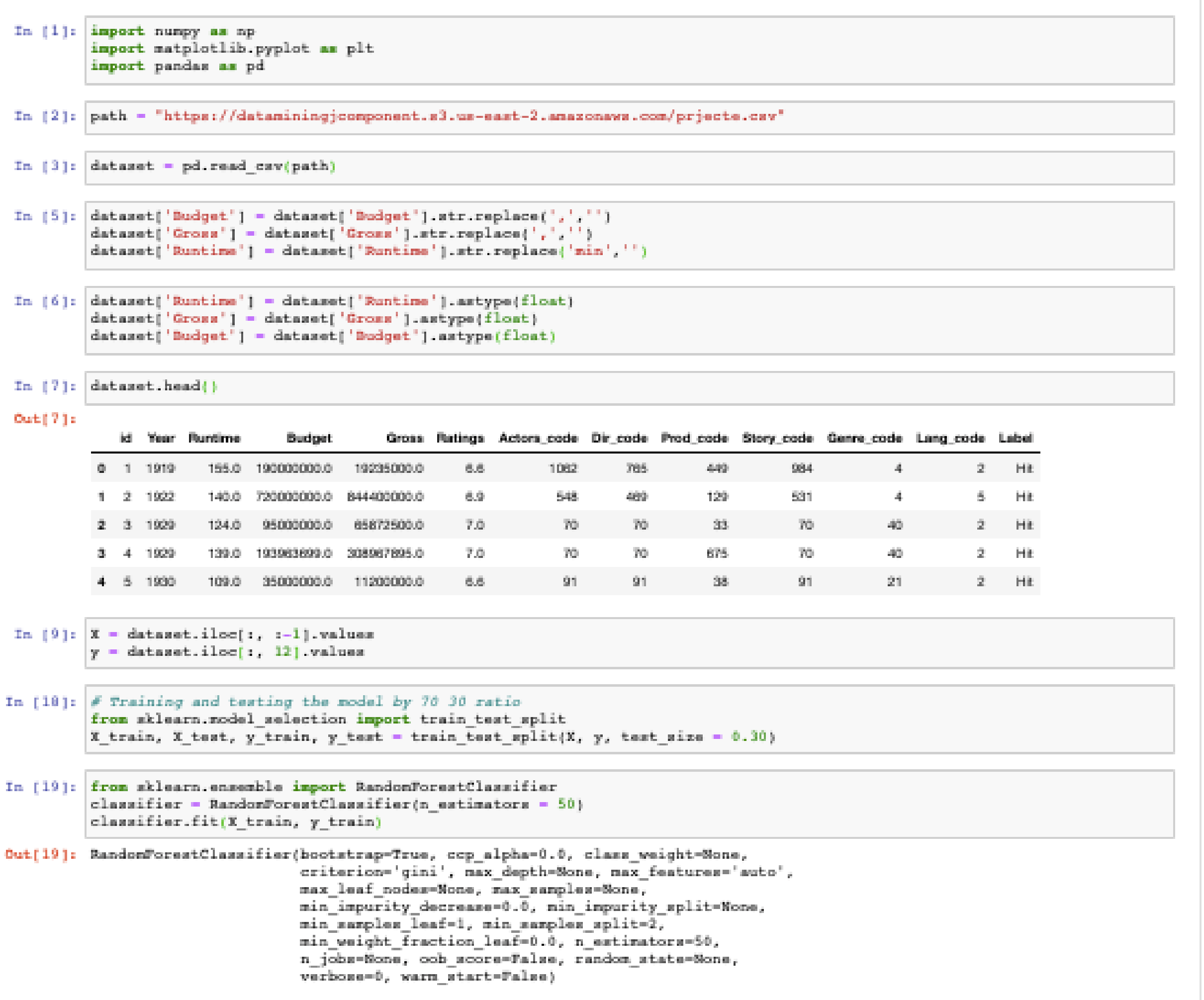


```
In [6]: feature_cols = ['id','Year','Runtime','Budget','Gross','Actors_code','Dir_code','Prod_code','Story_code','Genre_code','Lang_code'
        X = dataset[feature_cols] # Features
        y = dataset.Label # Target variable

In [7]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 1)

In [8]: clf = DecisionTreeClassifier()
        clf = clf.fit(X_train,y_train)

In [9]: y_pred = clf.predict(X_test)

In [10]: from sklearn.metrics import classification_report, confusion_matrix, accuracy_score
         result = confusion_matrix(y_test, y_pred)
         print("Confusion Matrix:")
         print(result)
         result1 = classification_report(y_test, y_pred)
         print("Classification Report:",)
         print (result1)
         result2 = accuracy_score(y_test,y_pred)
         print("Accuracy:",result2)

         Confusion Matrix:
         [[ 43  64   3]
          [ 48 183  15]
          [  1  13   2]]
         Classification Report:
                       precision    recall  f1-score   support

                 Flop       0.47      0.39      0.43       110
                  Hit       0.70      0.74      0.72       246
            Super Hit       0.10      0.12      0.11        16

             accuracy                           0.61       372
            macro avg       0.42      0.42      0.42       372
         weighted avg       0.61      0.61      0.61       372

         Accuracy: 0.6129032258064516
```



```
In [16]: from sklearn.tree import export_graphviz
         from sklearn.externals.six import StringIO
         from IPython.display import Image
         import pydotplus
         dot_data = StringIO()
         export_graphviz(clf, out_file=dot_data, filled=True, rounded=True,
             special_characters=True,feature_names = feature_cols,class_names=['Flop','Hit','Super Hit'])

         graph = pydotplus.graph_from_dot_data(dot_data.getvalue())
         graph.write_png('Pima_diabetes_Tree.png')
         Image(graph.create_png())

Out[16]:
```

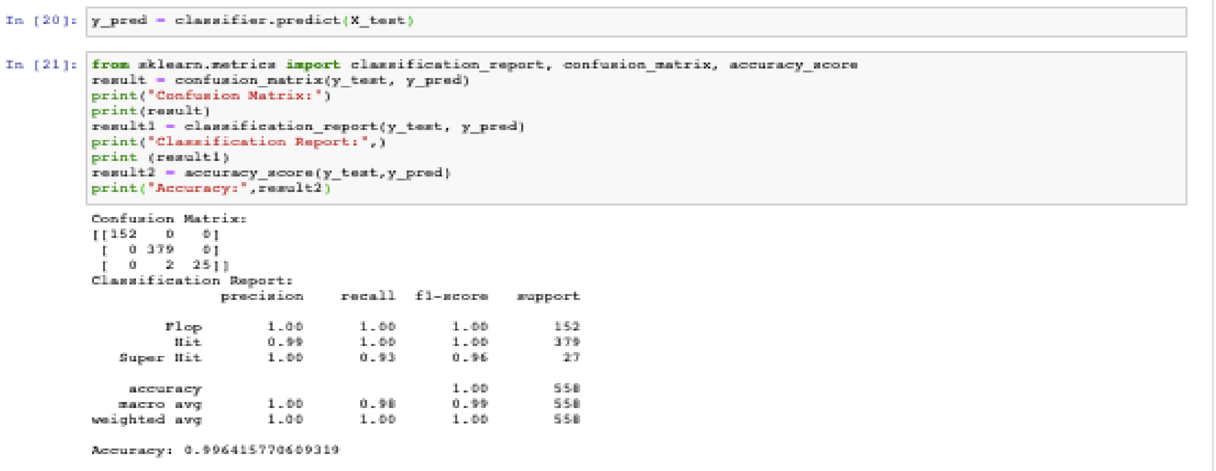
APPLICATION OF RANDOM FOREST ALGORITHM



```
In [1]: import numpy as np
        import matplotlib.pyplot as plt
        import pandas as pd

In [2]: path = "https://dataminingjcomponent.s3.us-east-2.amazonaws.com/prjects.csv"

In [3]: dataset = pd.read_csv(path)

In [5]: dataset['Budget'] = dataset['Budget'].str.replace(',','')
        dataset['Gross'] = dataset['Gross'].str.replace(',','')
        dataset['Runtime'] = dataset['Runtime'].str.replace('min','')

In [6]: dataset['Runtime'] = dataset['Runtime'].astype(float)
        dataset['Gross'] = dataset['Gross'].astype(float)
        dataset['Budget'] = dataset['Budget'].astype(float)

In [7]: dataset.head()

Out[7]:
```

	id	Year	Runtime	Budget	Gross	Ratings	Actors_code	Dir_code	Prod_code	Story_code	Genre_code	Lang_code	Label
0	1	1919	155.0	190000000.0	19235000.0	6.6	1062	765	449	984	4	2	Hit
1	2	1922	140.0	720000000.0	844400000.0	6.9	548	469	129	531	4	5	Hit
2	3	1929	124.0	95000000.0	65872500.0	7.0	70	70	33	70	40	2	Hit
3	4	1929	139.0	193983699.0	308987895.0	7.0	70	70	675	70	40	2	Hit
4	5	1930	109.0	35000000.0	11200000.0	6.6	91	91	38	91	21	2	Hit

```
In [9]: X = dataset.iloc[:, :-1].values
        y = dataset.iloc[:, 12].values

In [18]: # Training and testing the model by 70 30 ratio
         from sklearn.model_selection import train_test_split
         X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.30)

In [19]: from sklearn.ensemble import RandomForestClassifier
         classifier = RandomForestClassifier(n_estimators = 50)
         classifier.fit(X_train, y_train)

Out[19]: RandomForestClassifier(bootstrap=True, ccp_alpha=0.0, class_weight=None,
                                criterion='gini', max_depth=None, max_features='auto',
                                max_leaf_nodes=None, max_samples=None,
                                min_impurity_decrease=0.0, min_impurity_split=None,
                                min_samples_leaf=1, min_samples_split=2,
                                min_weight_fraction_leaf=0.0, n_estimators=50,
                                n_jobs=None, oob_score=False, random_state=None,
                                verbose=0, warm_start=False)
```



```
In [20]: y_pred = classifier.predict(X_test)

In [21]: from sklearn.metrics import classification_report, confusion_matrix, accuracy_score
         result = confusion_matrix(y_test, y_pred)
         print("Confusion Matrix:")
         print(result)
         result1 = classification_report(y_test, y_pred)
         print("Classification Report:",)
         print (result1)
         result2 = accuracy_score(y_test,y_pred)
         print("Accuracy:",result2)

         Confusion Matrix:
         [[152   0   0]
          [  0 379   0]
          [  0   2  25]]
         Classification Report:
                       precision    recall  f1-score   support

                 Flop       1.00      1.00      1.00       152
                  Hit       0.99      1.00      1.00       379
            Super Hit       1.00      0.93      0.96        27

             accuracy                           1.00       558
            macro avg       1.00      0.98      0.99       558
         weighted avg       1.00      1.00      1.00       558

         Accuracy: 0.996415770609319
```

RESULTS & CONCLUSION

After testing both the algorithms on the IMDb dataset i.e. Decision Tree and Random Forest algorithm, we found that the Random Forest algorithm got a better accuracy (99.6%) on the data rather than decision tree algorithm in which we obtained just 60% accuracy.

REFERENCES

1. Saraee, M., White, S., & Eccleston, J. (2004). A data mining approach to analysis and prediction of movie ratings. *WIT Transactions On Information And Communication Technologies*, *33*.

2. Chang, B. H., & Ki, E. J. (2005). Devising a practical model for predicting theatrical movie success: Focusing on the experience good property. *Journal of Media Economics*, *18*(4), 247-269.

3. Sharda, R., & Delen, D. (2006). Predicting box-office success of motion pictures with neural networks. *Expert Systems with Applications*, *30*(2), 243-254.

4. Abel, F., Diaz-Aviles, E., Henze, N., Krause, D., & Siehndel, P. (2010). Exploiting the blogosphere to forecast profit of music and movie products. *Technical report, L3S Research Center*.

5. Viana, W., Santos, P. O., da Silva, A. P. C., & Moro, M. M. (2014, October). A Network Analysis on Movie Producing Teams and Their Success. In *2014 9th Latin American Web Congress* (pp. 68-76). IEEE.

6. Chaudhari, N., Saini, M., Kumar, A., & Priya, G. (2016, December). A Review on Attribute Based Encryption. In *2016 8th International Conference on Computational Intelligence and Communication Networks (CICN)* (pp. 380-385). IEEE.

7. Lash, M. T., & Zhao, K. (2016). Early predictions of movie success: The who, what, and when of profitability. *Journal of Management Information Systems*, *33*(3), 874-903.

8. Ahmad, J., Duraisamy, P., Yousef, A., & Buckles, B. (2017, July). Movie success prediction using data mining. In *2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)* (pp. 1-4). IEEE.

9. Mahmud, Q. I., Mohaimen, A., & Islam, M. S. (2017, December). Marium-E-Jannat,"A Support Vector Machine mixed with statistical reasoning approach to predict movie success by analyzing public sentiments". In *20th International Conference of Computer and Information Technology (ICCIT)* (pp. 22-24).

10. Quader, N., Gani, M. O., Chaki, D., & Ali, M. H. (2017, December). A machine learning approach to predict movie box-office success. In *2017 20th International Conference of Computer and Information Technology (ICCIT)* (pp. 1-7). IEEE.

11. Ahmad, J., Duraisamy, P., Yousef, A., & Buckles, B. (2017, July). Movie success prediction using data mining. In *2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)* (pp. 1-4). IEEE.

12. Wijekoon, A. W. M. K. S. A., Sandanayake, T. C., Jayawardena, K. D. A. A., Buddhini, A. L. Y., & Ariyawansha, U. K. D. G. S. (2017, December). Spatio-Temporal Visualization Model for Movie Success Prediction Based on Tweets. In *Proceedings of the 2017 International Conference on Information Technology* (pp. 227-231).

13. Ahmad, P. Duraisamy, A. Yousef and B. Buckles, "Movie success prediction using data mining," *2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Delhi, 2017

14. Magdum, S. S., & Megha, J. V. (2017, June). Mining online reviews and tweets for predicting sales performance and success of movies. In *2017 International Conference on Intelligent Computing and Control Systems (ICICCS)* (pp. 334-339). IEEE.

15. Parikh, Y., Palusa, A., Kasthuri, S., Mehta, R., & Rana, D. (2018). Efficient word2vec vectors for sentiment analysis to improve commercial movie success. In *Advanced Computational and Communication Paradigms* (pp. 269-279). Springer, Singapore.

16. Çizmeci, B., & Ögüdücü, Ş. G. (2018, September). Predicting IMDb ratings of pre-release movies with factorization machines using social media. In *2018 3rd International Conference on Computer Science and Engineering (UBMK)* (pp. 173-178). IEEE.

17. Chakraborty, P., Rahman, M. Z., & Rahman, S. (2019). Movie Success Prediction using Historical and Current Data Mining. *International Journal of Computer Applications*, *975*, 8887.