A Prediction System for Movie Revenue

Chi-Ju Wu
Computer Science
University of Colorado Boulder
Boulder Colorado United States
Chiju.Wu@colorado.edu

Yu-Lin Chou
Computer Science
University of Colorado Boulder
Boulder Colorado United States
Yu.Chou@colorado.edu

Ziying Zhang
Computer Science
University of Colorado Boulder
Boulder Colorado United States
Ziying.Zhang@colorado.edu

ABSTRACT

In modern society, the movie industry grows rapidly and the cost of movie production is getting higher. In this condition, production companies have to be careful about evaluating the value of a movie. We aim to design and implement a system to predict movies' revenue. There are some impressive prior works, but they merely combine the movie metadata and the data from social media. The proposed system aims to help production companies predict the revenue of movies. To be exact, we design and implement a comprehensive system with both movies' datasets and data from social media to predict movies' revenue. Based on TMDB datasets, we also adopt the Instagram hashtag data indicating the popularity of actors and directors. Next, we preprocess the whole datasets by cleansing data based on missing budgets or revenues. After preprocessing, we implement clustering techniques to cluster revenue. Last, we use classification approaches, including Naïve Bayesian, Decision Tree, and ensemble methods, to predict the revenue. According to our results, using Random Forest with AdaBoost is the best way to predict the revenue. Our proposed method can reach 80% accuracy in average and may be suitable to be a reference for production companies.

INTRODUCTION

The movie is an indispensable part of our daily life and the film industry also seems to be the area where the hot money most likes to join. But behind the glamorous appearance, making a movie is an uncertain and dangerous investment, the film industry actually has to face four kinds of problems:

First of all, even you collect the best team and invest enough money to produce, the rate, the time, the competitor in the same period, the hot topics in mass media, even an international political issue can influence its performance, such as A Quiet Place with a small budget and earned 10 times box office because it chose the unpopular time to release.

Secondly, the movie is more like an industrial product. More investment means more promise or confidence in quality. Thus, a movie, especially the commercial film like Marvel, is an unstoppable trend to have gradually increased huge investment. And more investment means more dangerous when the company cannot get their money back.

What's more, the production life cycle of a film is always very long. It typically can take a film one or two years from inception to distribution before it reaches the public. The long cycle of a film may easily make its result be influenced by multiple uncertain factors.

In addition, this is an age of social media. It's no doubt that there are easier channels to reach the end customers. With more followers on Instagram and more views on YouTube, you can easily draw more people to watch your movie. However, getting the positive promotion will be more difficult. 20 years ago, a positive review and score in NYT, Variety or Screen might influence most audiences' choices. But today, if there is a famous Twitter account giving a critical opinion, you may lose thousands even more potential audiences in a short time.

These problems triggered us to find the potential pattern for a successful movie. We want to consider different issues to see what factors may play the important roles to make this movie profitable.

There are many prior works to explore how to predict the success of a movie in both art and financial aspects. Lash, M. etc. [3] and his group provide an example model considering as many as possible factors from "who" "when" and "what" via IMDB and Box Office Mojo data. Saraee. M etc.'s research [1] proved that actors/actresses (90% relevant) in a film are the most important factors to its success. The others use the data mining from mass media and social media data in analysis. For example, Asur and Huberman [10] use the number of mentions in Twitter and the emotion analysis of tweet content to predict movie box-office trends. But users' preference in social media transfer quickly. E.g., Instagram has more active users than Twitter which makes it have more valuable information to study.

In this project, we build a prediction model for the movie's range of revenue, using multiple data attributes from the movie's metadata from TMDB and the most popular social media user community-Instagram to help the production companies predict the revenue of movies. Based on TMDB datasets. We design and implement a revenue prediction system which is trained by the history of movie data with the

same genre, director and actors, and the added Instagram hashtag data of the actors and directors, which indicates their popularity and helps us improve the accuracy. After the data preprocessing and integration, DBSCAN is used to remove the outliers and Naïve Bayesian, Decision Tree, Ensemble method including Bagging, Boosting and Random Forest, and Integrate Random Forest Classifier into Bagging and Boosting are applied to predict the revenue.

Through the comparison of the above methods in our project, we find that the best way to predict the revenue is using Random Forest with AdaBoost. The average accuracy (around 80%) proves that the idea to integrate metadata of movie and the popularity in Instagram can work. And the 7% improvement of the prediction accuracy after adding the hashtag data from Instagram indicates social media data has a positive influence in improving the predicted model effectively.

In the following parts, we introduce the details about how to implement this project: The "Related Work" reviews the related literature to do the movie prediction for both art and financial success, and excellent cases to use social media or mass media data to do the analysis. The "Methodology" introduces each step in the whole process to implement the prediction system, from the dataset choice, data preprocessing, data analysis and implementation of different classifications. The "Evaluation" shows the results of different classification methods and the comparison with or without social media data. The "Discussion" exhibits the reflections from the project and the future works we can continue to do. The "Conclusion" summarizes the project.

RELATED WORKS

The purpose of data mining for the movie usually focuses on the common assumptions about its future financial or art success.

For art success, the rate in the main online platform is the generic criterion to do evaluation. Saraee. M etc.'s research [1] analyses 390000 movie data from IMDB, which concentrates on attributes relevant to the user ratings of movies. Although they find that it is difficult to apply data mining techniques to the data in the IMDb because the data cleaning and integration need more time than predicted, their classifier result indicates the director (55% relevant) and actors/actresses (90% relevant) involved in a film are the most important factors to its success. The study from Ahmad, J. etc. [2] develops a new mathematical model to predict Bollywood Hindi movie's success. Besides, the basic variables of a movie, they also consider the competitors at that time, release locations and target audiences when building the model. Via X^2 analysis, they finally find the genre and actors determines the success rating of movies, and there is a strong correlation between actors and the certain genres as well. All of these conclusions emphasize the star's impact on a movie.

For financial success, researchers are interested in the metrics like box office, revenue, and the profitability etc.. The forecasting of these indicators is one of the most attractive parts for the real film industry participants. The study from Lash, M. etc. [3] about the prediction the profitability provides a great example considering as many influence factors as possible. They use the IMDB and Box Office Mojo data and define four features to influence the revenue: 1). "Who" includes the star power and the dynamic collaboration network among them dug by social network analysis and text mining techniques; 2). "When", the released time and the competitors; 3). "What", contains the metadata and text of movie plot synopsis; 4). "Hybrid" indicates the match rules between the above features. And their analysis system based on these features can recommend a set of profit-maximizing cast members. Nithin V.R. and his partners [4] use the IMDB, Wikipedia, and Rotten Tomatoes data to predict the movie box office. They combine the nominal and numeric attribution and apply a greedy backward algorithm under supervised learning technique to implement three kinds of model: Logistic Regression, SVM Regression, and Linear Regression. And their result that linear model performs are more accurately gives us a reference to help us choose the model method. As well as the rate of film, some empirical studies in economic about star impact [7] [8] [9] emphasize that the top stars are also the important factor to influence the revenue increase, even if artistic star power is lower than commercial star power [8]. In addition, methods in deep learning [5][6], can apply to the revenue prediction as well.

Besides, the movie platform data integrating with the data mining or analysis from mass media and social media becomes a more popular way to do the financial prediction these years. Asur and Huberman [10] is the first one who pay attention to the social media and movie revenue prediction. They focus on the tweet's emotion analysis in Twitter and count the number of mentions and the positive or negative comments to predict movie box-office trends. The research from Apala, K. R.'s team [11] uses multiple social media and web source data (e.g., the historical movie database, the number of followers on Twitter, and a YouTube viewers' comments) to do the data mining and contribute to the prediction of the box office. Liu T. etc. [12] indicates that there are some high revenue movies have a low rate in movie platform, and more positive reviews cannot automatically translate to more people watching the said movie in the cinema. Thus, they only pay attention to digging the number of users who express their intents to watch a specific movie on social media. Their social media analysis is not using the number of followers but the analysis of unstructured texts. In addition, Zhang and Skiena [13] consider the traditional mass media impact and combine the IMDB data and news data to predict movie box-office revenues. Their correlation analysis

are not only limited in the evaluation of the media coverage in movie titles, but also add the coverage in directors, top 3 actors, and top 15 actors. The research of Joshi, M al et. [14] only focuses on the film critics' reviews from several popular traditional media in the film area, like Variety. Their new dataset matches the movie reviews with metadata and revenue data and can predict the important opening weekend revenue.

However, the above works usually focus on Twitter and Facebook to dig the social media data. In these three years, Instagram developed very quickly. Based on the global social media ranking 2018, Instagram has one thousand million active users while Twitter only has only one-third of it. And compared to Twitter, Instagram is based on the photos and the short video, which is better for the movie propaganda. So we think Instagram would be a better platform to get the real movie popularity data among the potential audiences. Based on the famous actors and directors' popularity and trends in the social media community, we believe our result based on Instagram could be more precise and reliable.

METHODOLOGY

1. Problem formulation

As there are many attributes in the metadata of a movie and Instagram also has lots of data can be analyzed, the main problems for us are

- how to choose the correct attributes which are correlated to the revenue,
- how to choose an appropriate classification method to build the model.

Moreover, before the data training, we need to consider how to cleanse the initial data, how to integrate the data from the different resources, how to convert nominal ones into numeric ones, and how to remove the outliers.

2. Datasets

The initial dataset is 5000 movies from The Movie Database (TMDB) [15]. It includes almost all metadata attribute of a movie. We used both the nominal attributes and numerical attributes of this dataset. The adopted attributes are shown in the below Table 1.

Type	Attributes
Nominal	genres, keywords, original_language, original_title, production_contries, production_companies, spoken_languages, cast, crew
Numerical	budget, popularity, release_date, revenue, vote_average, vote_count

Table 1 Attributes in TMDB dataset

3. Preprocessing

3.1. Solve the missing values in dataset

3.1.1. IMDB Data

For the missing value side, since the original TMDB dataset has a lot of missing data in revenue, we decide to use Web Crawler to grab information from IMDB to solve this problem. In order to do that, first we need to get the IMDB ID of each movie since the IMDB URL is not the combination of string. And we use the OMDB API to solve it. The next part is getting genres and production companies information form IMDBpy API, we use python to post the IMDB ID to IMDBpy and then generate the corresponding object to integrate into our dataset for the future analysis. Then implemented web crawler with python to get the budget and the gross USA, while doing this problem, we find that the HTML form in IMDB is not always consistent, so we need to consider the different situation to avoid corner cases, and then get the correct data to integrate into our dataset.

3.1.2. *Delete missing value*

After analyzing and comparing, we find the web crawler data from IMDB also has some missing values in revenue, especially the old movies. What's more, the web crawler usually replaces such empty blank with the date and we have to check and revise it one by one. Thus, in the final, we give up the web crawler data from IMDB and cleanse the data of the initial dataset. We delete all movie with an empty value in budget or revenue, and finally, there are 3239 data left.

3.2. *Add new attributes to change the nominal attributes to numerical ones*

As some important attributes in the movie metadata are nominal ones, such as genre and actors, if we want to apply it in the predicted model, it is necessary to convert it into numerical ones. The method to do it in this project is to add new attributes calculated by the history data to represent the influence of people and genre. For example, for a specific actor, we calculate the sum of all the movies' revenues he participated and then divided by the sum of those movies' budgets to get a new attribute "actor_point". And for each movie, we add "genre_point" "director_point" and "actor_point" "company_point" as the new attributes.

3.3. *Social media data integration*

For measuring the popularity of leading characters in the movie, the proposed work uses the hashtag numbers of each main actor, actress and the most important one from other actors on Instagram to represent its popularity of characters instead of the number of followers. Plus, we also use hashtag

numbers of the director and the title of each movie to indicate the popularity among these two attributes. Since not every actor/actress has an Instagram account, we adopt the hashtag attribute to represent the popularity. As the consequence, based on each movie's cast in the TMDb dataset, we transform five attributes to fit the format of Instagram hashtag and acquire the information from Instagram website. After downloading all hashtags, we integrate these attributes for showing numbers of hashtags of the actor and actress.

4. Data analysis

4.1. Correlation analysis

To find out the main attribute influencing the revenue, we analyze the correlation of all attributes and revenue. For the integrated dataset, we make two rounds statistic cleaning tasks before the correlation analysis. The first round is to filter those with an empty value in budget or revenue in the data preprocessing, and the second round is to remove the outliers of Instagram hashtag number based on the result of Round 1.

Firstly, we calculate the max number of all three main actors' hashtag number, the max number of all actors' and director's hashtag number, the ratio of budget and revenue of each piece of data, the sum of all three main actors' hashtag number. Secondly, we set zero and different N*STD as the lower and upper standard of outliers respectively. And we calculated the correlation coefficient r between these numerical attributions and the correlation coefficient results show in Table 2

The correlation coefficient of revenue and other attributes				
N (AVG+/-N*STD)	1.5	2	2.5	3
Budget	0.5203	0.5494	0.5991	0.6314
Actor1_hashtag	0.2052	0.1779	0.1722	0.1868
Actor2_hashtag	0.1446	0.1167	0.1081	0.1309
Actor3_hashtag	0.1101	0.0671	0.0388	0.0641
Director_hashtag	0.0076	0.0784	0.0803	0.0879
Movie_hashtag	-0.0016	0.0061	0.0074	-0.0029
Sum of Actor Hashtag	0.2495	0.2063	0.1814	0.2118
Max of Actor Hashtag	0.2091	0.1686	0.1451	0.1671
Max of all Hashtag	0.2054	0.1686	0.1456	0.1667
Ratio of Revenue/Budget	0.0377	0.0291	0.0249	0.0183
Left data after cleaning	2598	2817	2979	3045

Table 2 The correlation coefficient of revenue and other attributes

From the result of the correlation analysis, we can observe two things: Firstly, the best correlation result appeared when n = 3, and it is the one deleted fewest data. So using the general statistic method to delete the outliers is not suitable, because it usually has a better result when deleting more outliers. It inspires us to give up the general statistic method to delete outliers but to replace it by DBSCAN to find and delete the outliers. Secondly, only the budget has a significant correlation with revenue. For other attributes, the best correlation result is the sum of actors' hashtags from Instagram. Thus, when building the model, we should consider the different correlation of these attributions and set different weight.

4.2. Descriptive statistic for genre and production company

For the genre and the production company, we do the descriptive statistics for all independent values and its revenues by a python program. For the genre and the production company, we do the described statistic for all independent values and its revenues by a python program. For the production company, there are too many companies, and most of them only have one piece of data. So, we give up considering this attribute and the new attribute company_point, because it is too easy to be affected by individual data. The results of different genres are shown in Figure 1. From it, we can observe the difference between among different genres, and each type of genre has enough data, so that it and the created attribute genre_point are meaningful for the revenue prediction.

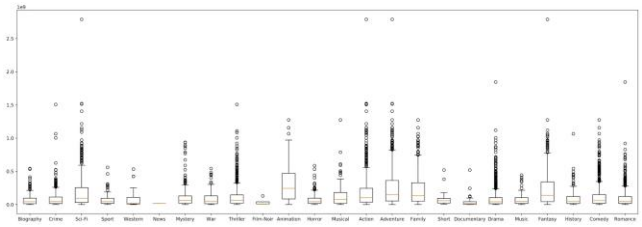


Figure 1 The box plot of different genres' revenue

4.3. Reduction of attributes

Based on the above analysis, we reduce the attributes which would be used for data training. Only budget, genre_point, the three actors' hashtag from Instagram, actor_point, the director's hashtag from Instagram, and director_point are kept, and other attributions unused in this model are deleted.

4.4. Remove the outliers – DBSCAN

As the general statistic method to remove outliers by average and the standard deviation is not fit for this case, we consider using the data mining method learned from the course to do the cluster analysis and outliers finding. Finally, we choose the density-based method DBSCAN. As the statistic method result indicates that we should not delete too much data from the

dataset, we set 50 as the reference number to delete outliers. For the integrated datasets which already cleanse the missing value in budget and revenue, we mainly analyze these seven attributes: actor_point, actor1_hashtag, actor2_hashtag, actor3_hashtag, director_point, director_hashtag, genre_point,. For each attribute, such as genre_point, to find the ε-neighborhood of core object p, we set the different radius ε and the number of a cluster n. Then we do the scan of the whole dataset following the rules to see how many clusters, and how many noisy points are not belonging to any clusters. These noisy points are possible the outliers we need to remove. For example, the genre_point, we scan the radius ε from 0.05 to 0.5 (+0.01 every time), and the number of a cluster n from 200 to 600 (+50 every time). After the 360 times scanning, the set (ε=0.46, n=300) with the noisy points near to 50 (57 points) is found and we would apply it to make a judgment whether a piece of movie data is an outlier in genre_point or not.

The results of DBSCAN in these seven attributes shows in Table 3:

	number of clusters	number of noise points	radius	numbers
Director_point	1	51	5.9	200
Actor_point	1	53	30	200
Genre_point	1	57	0.46	300
Actor1_hashtag	1	35	620000	200
Actor2_hashtag	1	50	400000	200
Actor3_hashtag	1	51	500000	250
Director_hashtag	1	56	94000	200

Table 3 DBSCAN result for the setup of radius and numbers

We follow these sets of each attribute to find out the outliers. However, we find that some of the outliers are true some are not. Such as Justin Bieber, he is really popular, and the hashtag number is very high. But like Rain, a Korean star, his name is also a general word for weather. So, the high hashtag should be deleted. Through the DBSCAN to filter the outliers and checked each one's meaning in Instagram one by one. Finally, we remove the movie data which include the actor hashtag Rain, Pink, Jr. , Ninja, Astro, Divine, Vanity and director hashtag McG.

5. Design

After data preprocessing, the design of the proposed system has three part. The first part is outlier detection and removing. We adopt DBSCAN, a density-based clustering algorithm, to detect outliers. To be exact, any points do not belong to any group, they are outliers and need to be removed. Next, the proposed method uses K-Means clustering to cluster the revenue column. In this part, we set K to 5, so revenue is split into 5 groups. The purpose is mainly for evaluation. In other words, we can test the correctness of our system by measuring the rate of classifying movies to correct cluster. Lastly, we generate the test set by splitting the dataset into training set and test set randomly and tried to find out the best classifier by implementing and comparing different classifiers. After several tests, we can determine the best classifier and complete the whole system. The diagram is shown in Figure 2.

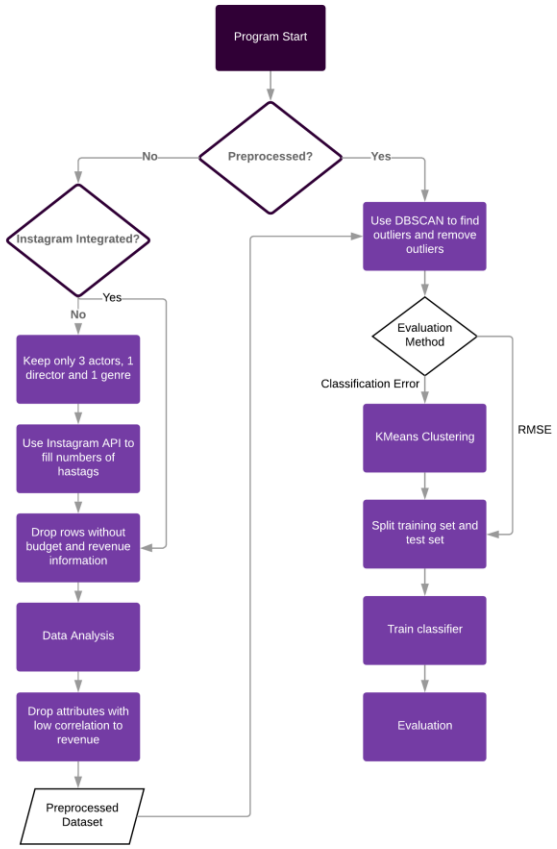


Figure 2 Flowchart of the whole system

6. Implementation

The methodology has 3 major components, these are: Single Classification (Decision Tree, Naive Bayesian); Ensemble method including Bagging, Boosting and Random Forest; Integrate Random Forest Classifier into Bagging and Boosting.

6.1. Single Classification (Decision Tree, Naive Bayesian)

In the beginning, we integrate the Decision Tree algorithm into our dataset after data cleaning and data integration, it is a non-parametric supervised learning method used for classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. Secondly, we try to use the

Gaussian Naive Bayes to do the single classification, it implements the naive Bayes training and classification algorithms for data that assumes that features follow a normal distribution.

	Decision Tree	Gaussian Naive Bayes	Random Forest	Bagging		AdaBoost		w/o Hashtags
base estimator				Decision Tree	Random Tree	Decision Tree	Random Tree	Random Tree
RMSE	0.131	0.175	0.087	0.113	0.0868	0.098	0.0706	0.0764
Classification								
avg correctness	0.75	0.2	0.79	0.74	0.81	0.79	0.82	0.75
avg diff	0.55	1.75	0.51	0.65	0.45	0.51	0.45	0.55

Table 4 Comparison between different classifier algorithm

The Formula (1) is the Gaussian Naive Bayes. After we compare the evaluation result in Table 4, we find that the Decision Tree Classifier has a better prediction result. So we decide to use decision tree as a base estimator in the bagging and boosting ensemble method.

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i-\mu_y)^2}{2\sigma_y^2}\right) \quad (1)$$

6.2. Ensemble method including Bagging, Boosting and Random Forest

In order to improve our prediction result, we use the ensemble method, which can use multiple learning algorithms to obtain better predictive performance. First one is Random Forest Classifier, a random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

The second one is Bagging, which involves having each model in the ensemble vote with equal weight. In order to promote model variance, bagging trains each model in the ensemble using a randomly drawn subset of the training set.

The third one is Boosting, we choose AdaBoost to be our boosting algorithm. An AdaBoost classifier is a meta-estimator that begins by fitting a classifier on the original dataset and then fits additional copies of the classifier on the same dataset but where the weights of incorrectly classified instances are adjusted such that subsequent classifiers focus more on difficult cases. Based on the result of Figure 9, we find that Boosting has the best prediction performance.

6.3. Integrate Random Forest Classifier into Bagging and Boosting.

From the resource [16], we find the concept provided by this thesis can help us to go deeper into our mining. After we use random forest classifier as a base estimator for AdaBoost, our model can combine random forest classifiers to obtain one stronger classifier and we find it creates the best traffic flow prediction based on Table 4.

EVALUATION

1. Data Set for training

The data set (training and testing set) which are used by this system will be produced from" TMDB", we finished data cleaning and data integration to do the mining.

2. Metrics

In this evaluation, we would split our original dataset into a training set and testing set. The training set size is 2917 and the testing set size is 258. After we trained, we implement two ways to test our conclusion in testing set. First one is RMSE calculated by Formula (2), it is a frequently used measurement of the difference between values predicted by a model and the values actually observed from the environment that is being modeled. RMSE serves to aggregate them into a single measure of predictive power. We also normalize the result of RMSE since we want to facilitate the comparison between different datasets and use the range (defined as the maximum value minus the minimum value) of the measured data as Formula (3).

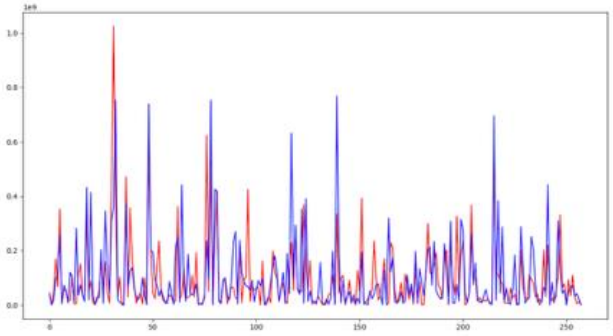
$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(X_{obs,i} - X_{model,i})^2}{n}} \quad (2)$$

$$NRMSE = \frac{RMSE}{X_{obs,\max} - X_{obs,\min}} \quad (3)$$

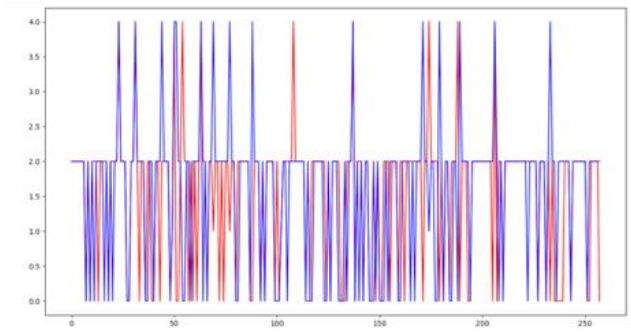
The second evaluation method is the classification error method. In the beginning, we use K-Means clustering to partition n observations into k clusters which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster. Then we do training and test the result in testing set to see whether it can fall in the right partition. We used two way to calculate our accuracy. First, we use average correctness, calculate the total correct prediction and divide by total testing number. The second one is average difference, we want to evaluate the difference between our prediction and correctness. If it falls into the wrong section, we calculated how far it is from the right section and add each point together then divided total data number to see the diff rate.

3. Methods to compare

In this section, we try several algorithms to evaluate our result. The first part is single classifier including Decision Tree and Gaussian Naive Bayes. Then we did ensemble method like Random Tree Forest and Boosting, Bagging. The last part we integrate Random Forest Algorithm to be the base estimator for Boosting and Bagging. Details of each Classifier result display from Figure 3 to Figure 9. The comparison between each result show in Table 4.

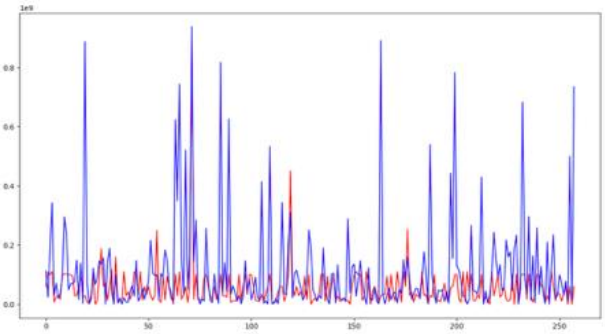


(a)RMSE

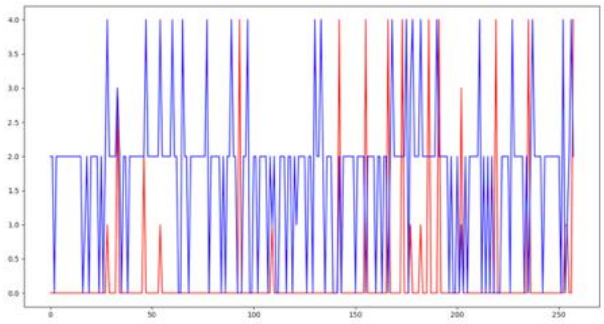


(b) Classification Error

Figure 3 Decision Tree Classifier Result

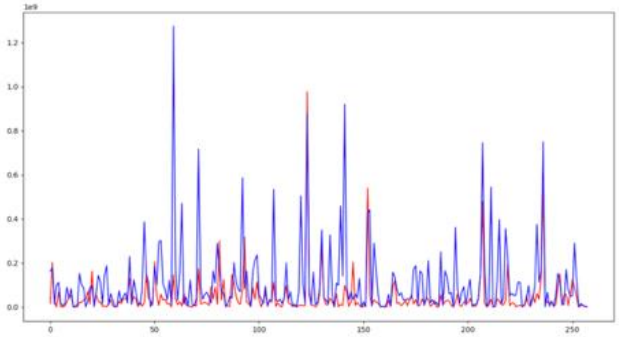


(a) RMSE

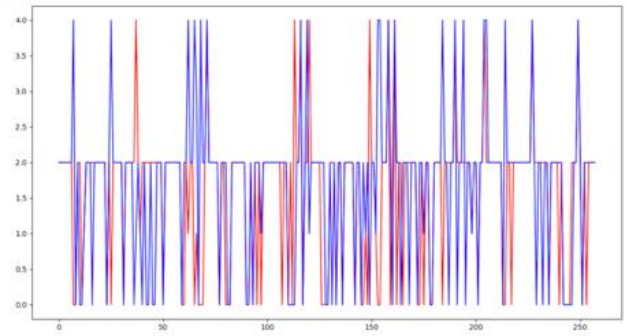


(b) Classification Error

Figure 4 Gaussian Naïve Bayes Classification Result

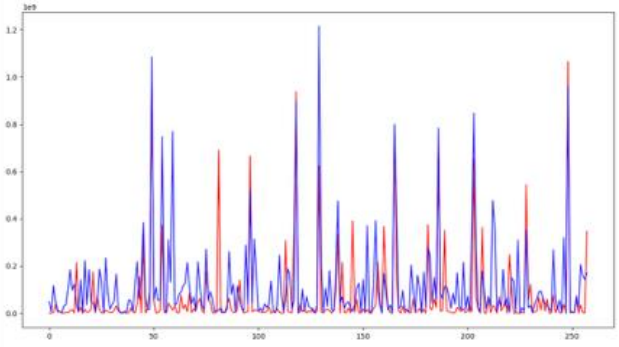


(a)RMSE

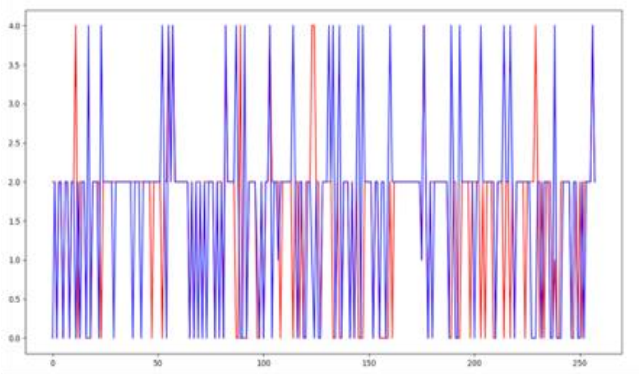


(b) Classification Error

Figure 5 Random Forest Classification Result

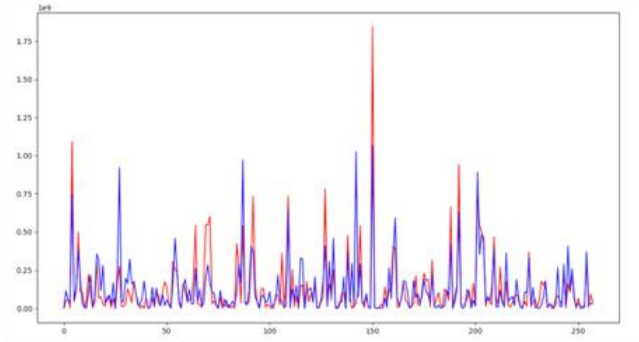


(a)RMSE

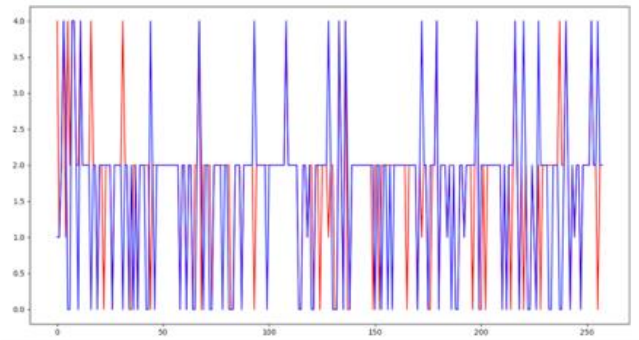


(b) Classification Error

Figure 6 Begging (with Decision Tree as base estimator) Result

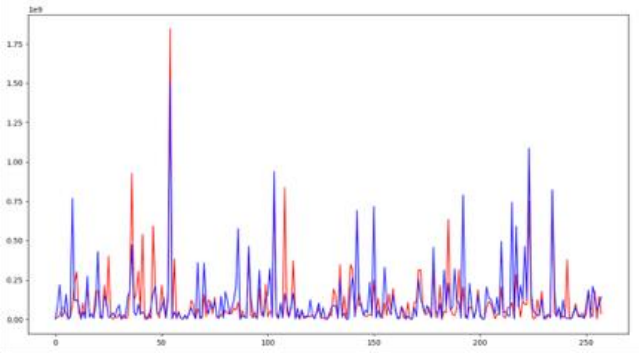


(a)RMSE

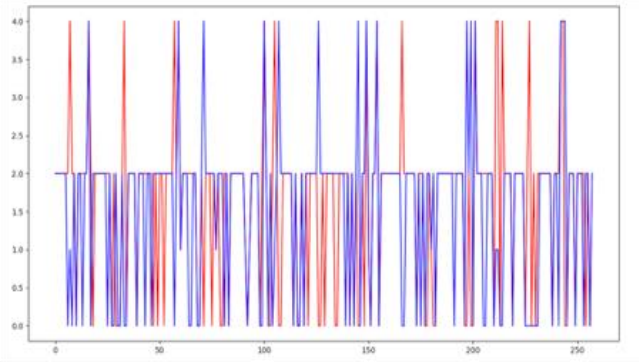


(b) Classification Error

Figure 7 Begging (with Random Forest as base estimator) Result

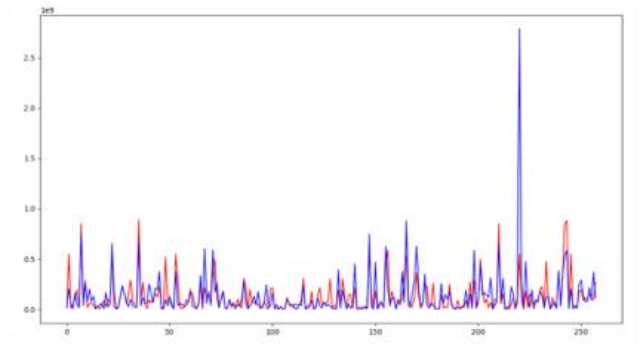


(a)RMSE



(b) Classification Error

Figure 8 Boosting (with Decision Tree as base estimator) Result



(a)RMSE

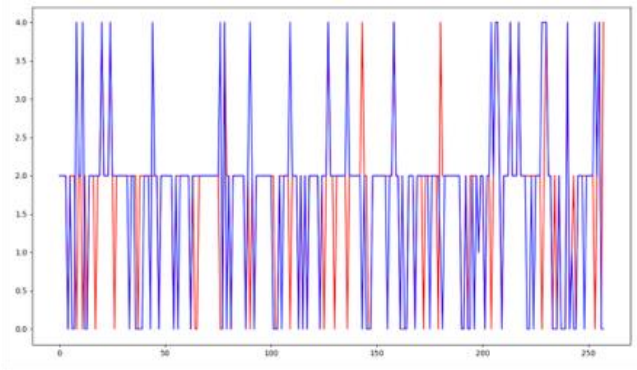


Figure 9 Boosting (with Random Forest as base estimator) Result

4. The effect of Instagram

In our research, we try to add a new attribute that have not been tested by prior work, which is Instagram hashtag. And we check whether this attribute plays an important role in the prediction analysis. In Table 4, once we remove the hash tag attribute and implement AdaBoost in our model, we find the accurate percentage would drop 7%, the RMSE performance would also be worse. It means social media can actually be a good reference while doing the prediction in movie revenue.

5. The best result and its explanation

For single classifiers, the accuracy of Decision Tree is significantly better than Naïve Bayesian Classifier. Because most of our attributes do not have duplicate values, we can infer the low possibility of each value. Hence, it is hard to get a precise prediction by Naïve Bayesian Classifier. On the other hand, based on the structure of generated Decision Tree, budget has the highest GINI index. It is the same as our results of data analysis; budget has the highest correlation to revenue. Once a Decision Tree adopts budget as its branches of the first layer, it is not far from a success. Therefore, we can have a concept about why Decision Tree classification is better than Naïve Bayesian classification. Next, as we know, bagging generally performs better than single classifier, because bagging will not be considered worse and is more robust to the effects of noisy data. The increased accuracy occurs because the composite model reduces the variance of the individual classifiers. Moreover, boosting performs better than bagging because boosting adopts weights to improve its performance. However, it risks overfitting the resulting composite model to training data. Our testing results perfectly fit the explanation above. Besides, we test a model using AdaBoost algorithm with Random Forest that this model has the highest accuracy. The Random Forest is essentially the combination of two independent ideas: bagging, and random selection of features. It can reduce the variance of each Decision Trees. Plus, as our results, using Random Forest with AdaBoost is better than all other methods. In addition, the generalization error for a forest converges as long as the number of trees in the forest is large. So, while the number of tree size increases, the accuracy also increases.

DISCUSSION

1. Lessons

In this study, we have some lessons to share. First, in the preprocessing stage, we reduce the field of actors to have only three main actors and store them in the comma-separated form “actor1, actor2, actor3”. It causes a lot of problems for us to split the string and find out the corresponding actor. We find it is better to store actors in individual fields. Next time, we will think more about the design of fields in datasets.

Second, we adopt DBSCAN algorithm to detect outliers. DBSCAN is a parameter-sensitive method. At first, we try to tune parameters manually, but it is hard to realize the current status by console logs and it will take too much time to find desirable parameters. Thus, we implement a visualization function to let us easily understand the results of DBSCAN. In addition, we use an automatic routine to tune the parameters until the detected outliers below a certain value. To conclude, it is always worth building a comprehensive program to deal with tuning parameters.

2. Limitation

The proposed method has two limitations. The first one is about the culture of using social media in different countries. Although Instagram has 1,000 million active users, it is still not a popular platform in some countries, like China. As a result, if we want to transform our result to another culture clusters, we need to consider more about the situations of Instagram.

Next, because of the limitation of Random Forest classification, the proposed method does not support incremental data. Once the datasets become larger, users need to spend a lot of time to train a new model while there are new movies.

3. Future works

The current model still leaves some works to improve the model in the future: The first one is to expand the dataset. The current dataset from TMDB only has 5000 pieces of movie data, and it leads that some attributes like production companies don't have enough information from each company to be used for the prediction model. And other datasets with more movies data don't have enough attributes as TMDB. So considering to improve the accuracy of the prediction model by machine learning, we need to expand it by the web crawler to fill in all attribute. The second one is the video view from YouTube. We

try to catch the video view information of the movie trailer, but a difficult problem is when we use the movie name to search it, there are lots of short videos about a film in the searching result such as the interview of actors or fan video. It is repeatedly appeared. Even for the trailer, there are many repeated ones by different uploaders and some uploaded one or two years after the movie launching. Thus, to catch the accurate video view of the movie trailer on YouTube and evaluate its influences before launching, we need to adjust each piece of movie data manually and avoid such disturb. It takes too much time. The third question is that we only collect Instagram's data as the social media data. Twitter, Snapchat, even TikTok can be considered to add in the future.

In addition, for the model, there are more methods can be explored to improve the result, such as Neural Networks. The comparison of the complexity and accuracy of each method and the rule of parameter adjustment are very valuable for building the related prediction model.

In the current stage, we cannot get temporal data from Instagram, so we used the latest number of hashtags for each actor and director. However, from a practical perspective, we cannot know the number of hashtags after movies being published. Therefore, it is better to acquire temporal data from Instagram instead of using the latest data in the future.

CONCLUSION

Nowadays, movie industry grows rapidly. The proposed system aims to help production companies predict the revenue of movies. To be exact, we designed and implemented a comprehensive system that it is capable to predict movies' revenue. Our idea is to integrate data from social media into a movie database and apply data mining techniques to create a robust movie revenue predictor. Besides the original TMDb datasets, we also adopt the Instagram hashtag data indicating the popularity of actors and directors. Next, we preprocessed the whole datasets by cleaning data with missing budget or revenue. After data integration and cleaning, we implemented DBSCAN to detect outliers and K-Means clustering to cluster revenue. In the next stage, we split the dataset into test set and training set and used classification approaches, including Naïve Bayesian, Decision Tree, and ensemble methods, to predict the revenue. According to our results, we noticed that ensemble methods are more accurate than single classifiers. Plus, we tried to use random forest with AdaBoost and got a good result. This algorithm took a lot of time to form a model; however, movies usually consume a lot of money and time, so we all think it is reasonable to pursue the best prediction by sacrificing some performance. Last, we recreated the model by removing the hashtag attributes to check the effect of social media. Based on the results, we can know that the accuracy of data containing hashtags is slightly greater than the accuracy of data without hashtags. It seems to be necessary to be improved, but we all think it is a decent start to integrate modern social media into movie datasets.

REFERENCES

[1] Saraee, M., White, S., & Eccleston, J. (2004). A data mining approach to analysis and prediction of movie ratings. Transactions of the Wessex Institute, 343-352.

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

[3] 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.

[4] Nithin, V. R., Pranav, M., Sarath, B., & Lijiya, A. (2014). Predicting Movie Success Based on IMDB Data. International Journal of Data Mining Techniques and Applications, 3, 365-368.

[5] Zhang, L., Luo, J., & Yang, S. (2009). Forecasting box office revenue of movies with BP neural network. Expert Systems with Applications, 36(3), 6580-6587.

[6] Sharda, R., & Meany, E. (2000). Forecasting gate receipts using neural network and rough sets. In Proceedings of the International DSI Conference (pp. 1-5).

[7] Nelson, R. A., & Glotfelty, R. (2012). Movie stars and box office revenues: an empirical analysis. Journal of Cultural Economics, 36(2), 141-166.

[8] Hofmann, J., Clement, M., Völckner, F., & Hennig-Thurau, T. (2017). Empirical generalizations on the impact of stars on the economic success of movies. International Journal of Research in Marketing, 34(2), 442-461.

[9] Liu, A., Liu, Y., & Mazumdar, T. (2014). Star power in the eye of the beholder: A study of the influence of stars in the movie industry. Marketing Letters, 25(4), 385-396.

[10] Asur, S., & Huberman, B. A. (2010, August). Predicting the future with social media. In Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology-Volume 01 (pp. 492-499). IEEE Computer Society.

[11] Apala, K. R., Jose, M., Motnam, S., Chan, C. C., Liszka, K. J., & de Gregorio, F. (2013, August). Prediction of movies box office performance using social media. In Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (pp. 1209-1214). ACM.

[12] Liu, T., Ding, X., Chen, Y., Chen, H., & Guo, M. (2016). Predicting movie Box-office revenues by exploiting large-scale social media content. Multimedia Tools and Applications, 75(3), 1509-1528.

[13] Zhang, W., & Skiena, S. (2009, September). Improving movie gross prediction through news analysis. In Web Intelligence and Intelligent Agent Technologies, 2009. WI-IAT'09. IEEE/WIC/ACM International Joint Conferences on (Vol. 1, pp. 301-304). IEEE.

[14] Joshi, M., Das, D., Gimpel, K., & Smith, N. A. (2010, June). Movie reviews and revenues: An experiment in text regression. In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (pp. 293-296). Association for Computational Linguistics.

[15] TMDB dataset: https://www.kaggle.com/tmdb/tmdb-movie-metadata

[16]Resource: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.109.224&rep=rep1&type=pdf&fbclid=IwAR019byviBfG-JgBzlgUfxgfz2sIRxMyoLP5dZEkv7dPSA4FR6Na2HHOE6I

APPENDIX

Honor Code

On my honor, as a University of Colorado Boulder student, I have neither given nor received unauthorized assistance.

Contribution

Chi-Ju Wu: Data preprocessing, Data clustering, Classification, Evaluation, Report
Yu-Lin Zhou: Data preprocessing, Classification, Report
Ziying Zhang: Data analysis, Data clustering, Report