Motion Picture Content Rating Predictor

Abhishek Jain, Anshul Grover, Anuj Bhambri, Rahul Chomal
17125618, 01726140, 17170222, 18104100
MSc. Data Analytics (Cohort-A)

26th April 2019

Abstract— **Motion Picture Association of America (MPAA) is an American film association founded in 1922 by motion picture producer and distributor of America. The sole objective of this association is to classify the movie content as per the guidelines. This project is a step towards developing a machine learning based classification system that classifies the MPAA rating of movies by learning the patterns from legacy data of movies, such as the name of the actors, the director, sentiments of people, box-office collection and more. The goal is to train machine learning models like ANN, Random Forest, Ada-Boost, Gradient Boosting, SVM and others on the Movie Industry dataset. The dataset is trained to classify the MPAA rating of the movies and subsequently tested on test data. ANN achieves the highest classification accuracy. Additionally, a new target variable is created, "Family Movie" to classify if a movie is a family movie or not. This binary variable achieves better classification accuracy when compared with the classification accuracy of the MPAA rating. The methodology used is promising, and various dimension reduction techniques like information gain and PCA is used in this project with K-cross validation to train and test each observation of data. The initial data in hand did not have enough information to classify the target variable which were tackled by extracting data from popular websites such as Youtube and Twitter adding more information to the dataset.**

Keywords—Machine learning, movies, classification, Random Forest, ANN, ADA-Boost, Naive Bayes, Decision Tree, Support Vector Machine, C5.0, Gradient Boosting, PCA (Principal Component Analysis), Information Gain, K-cross Validation;

I. INTRODUCTION

The recommendation systems are widely used in almost everything now, be it shopping websites like Amazon or watching a movie on a website such as Netflix. In recent years, the movie business has grown exponentially. The movie industry in the USA generated close to $43 billion in 2018 with an increase of 2.2% from the preceding year. It is also believed to be growing faster than the economy of the country. However, the attendance at the movie theatres shown a downfall with the increase in the rise of access to home entertainment options affecting the movie theatre outings. The movie distributors are shifting and relying heavily on domestic licenses including online streaming and on-demand videos. Another hassle the movie industry is going through is the rating that they need to get from the Motion Picture Association of America (MPAA). The MPAA rating is designed such to restrict the audience based on the content of the movie. For instance, a PG-13 rated movie may be seen by children above 13 but may contain scenes of aggression that can be problematic for the children. To get the rating, films are submitted to the board, and the producers pay a fee to get it rated. This research focuses on predicting the film's MPAA rating using various machine learning based classification models such as Random forest, ANN, Naïve Bayes and more. This paper also evaluates and compare the different machine learning algorithms to find out which classification-based machine learning model predicts the desired results efficiently. The motivation behind the research is the growing impact of predictive analytics. Also, the model can be of high significance to the producers of the movie as by using the model they will be to predict the MPAA rating of the film which can be helpful to them. For example, Eyes Wide Shut, a 1999 drama movie got a rating of NC-17(Adults only) by the Motion Picture Association of American (MPAA), but the director of the film digitally edited a scene so that the movie gets an R(Restricted) rating and the movie went on making $104,267,443, internationally. This model can also help the filmmakers to predict the film rating as a movie with explicit content and ratings such as **R** is released on a lesser number of screens than a **G** rated movie which attracts more mass and hence adding more to the overall revenue of the film.

The paper focuses on answering the following research question:

"Can classification-based machine learning models predict the MPAA rating of a movie?"

The research question formulated can be useful to the filmmakers that put so much effort in making the films and the films go through a process of getting a rating from the board. The board, if finds the movie to be unfit for a rating assigns another rating to the movie that can affect the overall business of the movie. The film censor board in the USA cannot mainly edit or censor a film, according to the first amendment to the US constitution; however, in India, there is a different scenario altogether. The board that approves the film certificate (CBFC) refused to give a release certificate to many films as the films were supposedly "hurting" the sentiments of a caste group. Therefore, this research will help the directors of the movie to rate the film before submitting it to the board and saving it from their axe. The MPAA can also use the model to classify the movies that are sent to them for rating. They can just put in the data, and the machine learning model can rate the movie, saving the time and hassle the committee goes through. The following sections will highlight the related work, the methodology, the implementation and the results achieved by the various machine learning models.

II. RELATED WORK

There are many different machine learning models implemented to predict the movie's rating based on various features like social media, blogs and news articles. There has been limited research in this field by using social media sentiment analysis (Twitter and YouTube) for movie rating prediction.

The researcher [1] has taken a similar approach in which the box office revenue and success of the movie was predicted by using a unique method of incorporating social media like Twitter and YouTube comments for the prediction. This research is based on a similar approach of using social media sentiment score with the Kaggle movie dataset [2] to predict the MPAA rating of the movie.

Research [3] has made a comparison of different models like Decision Trees, SVM, Random Forest, boosting techniques, etc. for predicting movie rating. The research has done feature transformation for 69 dimensions in the dataset. It was depicted that the change helps in improving the accuracy of the model significantly. Having a proper feature selection technique will help in building better models. The researcher [4] has implemented a feature selection named Information Gain (IG) in which a probabilistic model is used to predict the features with high dependencies on the dependent variable. Finding out the weight of the significant contributor and similarly, the least for the model and are removed from the model. [5], [6] proposed the comparison of different feature selection techniques and was evaluated that for a classification model with smaller dataset IG outperform other methods. The comparison of the accuracy of naive base used on similar dataset gave a considerable improvement of around 10% [7]. Hence the current research uses IG techniques for feature selection.

Research done in [8] has shown that the traditional regression model like logistic and simple logistic can be used for the predictive model with proper data transformation selection process. The model was developed to classify five different movie rating and used around 12 features. It was —shown the logistic regression outperforms all other models by 4% accuracy increase. Similar research done by [9] has predicted the revenue of the movie and the IMDB rating using the regression model and SVM. The dataset used in this research was social media dataset merged with featured movie dataset. The results showed that the regression model performs better when compared to simple SVM. It had an accuracy of around 7% better when the parametric tuning was not used in SVM and error tolerance of approximately 0.2 in both the models.

The author in [10] has built seven different classification model which ran on Linked Open dataset and combining it with reviews of the album. The models which were trained in this approach where linear SVM, KNN, RBF SVM, Decision Trees, Random Forest, AdaBoost, and Naive Bayes. The hyperparameter tuning used for this model uses a grid approach in which the values were incremented and tested, and best resulted in models' benefits were used as the final value for the hyperparameter. The result showed that the SVM model outperformed all other model and had an accuracy of 90%. This research showed that the hyperparameter tuning had a significant impact on the accuracy of the model since the analysis in [9] suggested a different result. A similar combined model approach was also made in [3] which has also included boosting techniques and Neural Networks (NN) for comparison. The results showed that boosting techniques like Adaboost and XGBoost had a similar level of accuracy when compared to SVM or NN. However, the hybrid NN models outperformed all other.

The research [4] has used IG for feature extraction and had a comparison of different models like Logistic Regression, Neural Network Classifier, Naive Bayes, decision tree. The accuracy of logistic regression performed better when compared to Naive Bayes, decision tree, etc. however the model performed marginally better with about 3% increase in accuracy. The models were evaluated by using Precision and recall and accuracy. The next paragraph will address the related work in the field of movies and the gaps that this research fills.

In the paper [11] the researchers developed a dataset of movies, *Right Inflight* which can be used by the researchers to predict whether the video is suitable for watching in a situation. The video preference was based on the data from the opinion of people by crowdsourcing the data and collecting the information on whether the participants will watch the video in a situation and the reason behind that. The researchers by collecting the data produced a list of 308 movies that the viewers will like to watch when on a flight. The idea behind focusing on the flight-entertainment system was to enhance the customer experience as the viewers will want to watch a movie to relax despite the uncomfortable fight. However, the shortcoming of using the dataset in this research is first the dataset is small, and some machine learning models require the right amount of data to evaluate correctly. Another shortcoming of the dataset is that the data is qualitative and splitting the data into suitable and not -suitable differs on a personal choice. The researchers in [12] formulated research to predict the crowd opinion of a movie before the movie is released. In the paper, they regarded the crowd opinion as Label Distribution Learning (LDL) problem and proposed an algorithm using Support Vector Regressor named as LDSVR to fit the function of each variable of the distribution by various output SVM. They found out that the algorithm proposed LDSVR predicted the crowd opinion accurately based on the metadata available about the movie. The paper does not discuss the viability of the data collected from various people and cannot be considered that proofs the rating as accurate.

The research done in [13] has used data mining techniques to mine trailers data of movies from YouTube and predicting how will they perform at the box-office. The authors used KNime for data analysis and applied supervised learning. Regression-based machine learning algorithms were used to predict the target variable, the gross income of the movie. The researchers found that Linear regression and Gradient boosted tree performed well in predicting the gross income with R^2 value of 0.891 and 0.889 respectively.

III. DATASET DESCRIPTION AND PREPARATION

This paper uses different kinds of data of movies taken from a few sources. The Movie industry dataset is taken from www.kaggle.com [2]. The dataset initially comprised of data of 6820 films from the year 1986-2016. The dataset has many attributes of a film, for example, the budget of the movie, the country of origin, the director of the movie, the name and so on. A table below will make it simple to understand the various attributes of the movies. A simple correlation test was done on the Kaggle dataset, and no significant relationship was found. Hence, the dataset was expanded and made unique by extracting data from YouTube and Twitter. Sentiments about the movie were derived from Twitter API and the likes, dislikes and the view count on the trailer of each movie on YouTube were extracted from YouTube's API using RStudio. This is discussed in the following paragraphs in brief.

Table 1: Data Description

Attribute	Description
Budget	The budget of the movie
Company	The production company
Country	Country of origin
Director	The director
Genre	Main genre of the movie.
Gross	Revenue of the movie
Name	Name of the movie
Rating	Rating of the movie (R, PG, etc.) [**Target Variable**]
Released	Release date (YYYY-MM-DD)
Runtime	Duration of the movie
Score	IMDb user rating
Votes	Number of user votes
Star	Main actor/actress
Writer	The writer of the movie
Year	Year of release

A. *DATA EXTRACTION FROM TWITTER*

As we all know, Twitter is a gold mine for data. Therefore, to expand the dataset and to follow a novel approach, sentiments about each movie were extracted from Twitter using the Twitter API in Python which allows the user to extract every tweet about a topic. The first step in extracting data from Twitter is to import the libraries. The python libraries used in extracting twitter sentiments are: tweepy, pandas and textblob, which are attached with the code in another file. The next step is to create a developer account on twitter, which was done before the extracting process. The twitter credentials are passed which includes the: consumer_key='xxxxx',consumer_secret='xxxx',access_key ='xxxx' and access_secret = 'xxxx'. Once these credentials are passed, the credentials are passed to Tweepy's *OAuthHandler* instance, **auth**, that calls a method, set_access_token in which the above created access_key and access_secret are passed. Once this is done, we pass the hashtag or the handle name (in this case, name of the movie) and the tweets are extracted. The extracted tweets needed cleaning, for example, the emojis and the special characters were removed by using the preprocessor library of python.

Since, Twitter is not so good with historical data, hence the data of 2378 movies were extracted from the Twitter API. The sentiments were categorized as follows:

- Positive sentiment = 1
- Negative sentiment = 2
- Neutral sentiment = 0

Once the data was extracted from the website, the gathered data was then written into a .csv file and was then combined with the clean master dataset using RStudio. There was a constant effort from the researchers of the paper to extract the geo-location of the tweets; however, not every Twitter user allows access to their location. This was another effort to expand the dataset but was not successful due to access constraints.

B. *DATA EXTRACTION FROM YOUTUBE*

As the data of 2378 movies were extracted from Twitter; therefore, data of the same movies were extracted from YouTube using tuber package in RStudio. The package allows to view the channel name, the number of subscriptions on the channel, views, likes, dislikes and the comments on the video. The following steps were implemented to extract data from YouTube:

- The Name of the movie was taken from the original dataset.
- The Youtube URL for searching the movie was appended with the movie name recursively.
- The first link on the result page was taken as the "youtube ID" using rvest and purrr package in R.
- To reduce the change of blocking our IP from google the process was delayed using system sleep command to make it more humanly. The list was then stored in a CSV file.
- Youtube developer account was created from which the app_id and authentication token were taken.
- The dataset which contains the youtube id was reloaded and the ID was used to extract the likes, dislikes, view count from the video using tuber and httpuv package in R.
- To make sure that the IP is not blocked the process was delayed using the system sleep command.
- The data extraction code can also handle the missing values and replaced it with 0 for likes and view count to make sure the extracted data is clean with no missing values.

To expand the dataset, the following data were extracted from YouTube which the researchers thought would be beneficial for the analysis and prediction:

- YouTube Likes: Number of likes on the trailer
- YouTube Dislikes: Number of dislikes on the trailer
- View count: Number of views on the trailer of the movie.

The extracted data was then combined with the master dataset and was ready for analysis. The final data included the data obtained from YouTube and Twitter along with the attributes from Table 1. The added columns were as follow:

Table 2: Extracted Attributes from Twitter and YouTube

Attribute	Description
Twitter Sentiments	Sentiments from Twitter: Positive, Neutral and Negative
YouTube likes	Number of likes on the movie trailer
YouTube Dislikes	Number of dislikes on the movie trailer
View Count	Number of views on the trailer of the movie

The goal of the project is to predict the MPAA rating of the movie based on various attributes of the movie. The MPAA rating is classified as follows according to the Motion Picture Association of America.

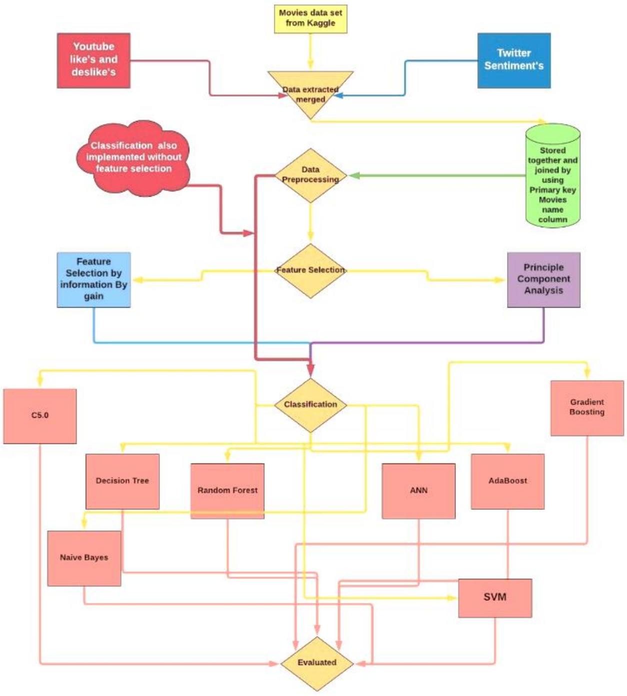
Table 3: MPAA Ratings *[14]*

Rating Symbol	Meaning
G	**General Audience**. All ages admitted.
PG	**Parental Guidance Suggested**. Some Materials may not be suitable for children
PG-13	**Parents Strongly Cautioned**. Some materials may not be suitable for children.
R	**Restricted**. Under 17 requires accompanying parent or adult guardian
NC-17	Adults only. No one 17 and under permitted.

To make better classification and simplify things, the ratings G, PG and PG-13 were classified as **Family Movies** and the rest two ratings, namely, R and NC-17 were classified as **Non-Family movies** to make the classification **binary**.

The dataset is hence ready to apply the machine learning models and get the results. The evaluation and results are discussed in the next section.

IV. METHODOLOGY



Movies data set from Kaggle
Youtube like's and deslike's
Twitter Sentiment's
Data extracted merged
Classification also implemented without feature selection
Stored together and joined by using Primary key Movies name column
Data Preprocessing
Feature Selection by information By gain
Feature Selection
Principle Component Analysis
Classification
C5.0
Gradient Boosting
Decision Tree
Random Forest
ANN
AdaBoost
Naive Bayes
SVM
Evaluated

Figure 1: Workflow of the Project

This project follows the Cross-Industry standard process for data mining (CRISP-DM) methodology. This data mining process allowed the researchers of the project a complete blueprint to carry out this research. The method mainly focuses on six phases which will be discussed in detail now with the context of the research. The first phase of the CRISP-DM methodology is the **business understanding** of the project that was kept in mind when building and implementing the models to fulfil the business needs of the project. As stated earlier, the project can be useful to both the filmmakers and the Movie Association. The business objective of the project was designed, and the knowledge was then converted into a data mining problem. By using feature extraction, the critical factors of the project were uncovered. The second phase as the methodology states is the **data understanding** of the project. The first step is to collect the data and then to get familiar with the data. In this research, the data was collected from www.kaggle.com, and the data was expanded from YouTube and Twitter. The data extraction process is described in the section above. Getting familiar with the data involved checking the data volume and examining the properties. To understand the data better, the researchers used the CRISP-DM methodology to check the attribute types, the range of each attribute, correlations and the identities of the data.

Further data analysis was done on the final dataset to carry out the exploratory data analysis and properties of each interesting attribute was analysed in detail. All the missing values were removed, checking the plausibility of the values, for instance checking whether all the fields have the same or nearly equal values. The next phase is the **data preparation** of the final dataset from the raw data. In this project, the final dataset was built by combining the Movie Industry dataset from Kaggle with the twitter sentiments and the number of likes, dislikes and the view count on the trailer of the movie. The final dataset constituted the data of 2378 movies with 19 features including the target variable, the rating of the film. The next phase according to the CRISP-DM methodology is the **modelling** phase of the project. In this phase, various machine learning models were selected and applied by calibrating the parameters to optimal values. The models chosen for the research are the classification-based machine learning models, namely, Naïve Bayes, c5.0, Random Forest, Decision tree, ANN, SVM and boosting algorithms such as ADAboost classifier and gradient boosting. The next step is the **evaluation** part of the project. To reach the desired business objective, all the models need to be thoroughly evaluated. The outcomes of each model and the feature extraction are discussed in **section VI** of this report. The final phase of the research is the **deployment** of the project. This phase focuses on implementing the results and how to utilise them according to the need of the customer. Since this is an academic data mining project, the deployment here refers to generating of this report to prove the functionality of the data mining steps that are covered in the course of this research.

V. IMPLEMENTATION

A. DATA EXTRACTION

Classification of movie rating is implemented on the original data set extracted from www.kaggle.com, but the classification accuracy achieved was between 50 and 55% depending on the model. So, these results proved that the original data set did not contain any relevant pattern information for our target variable. Therefore, additional data were extracted from the two popular social networking websites, YouTube and Twitter. This is done to get more data which have the relevant information for classification of our target variable, MPAA rating, to develop a machine learning model to predict the Movie rating by analysing the legacy data. This project is first of its kind developed from scratch, and no work is done before related to this, hence an opportunity to evaluate as many machine learning models as possible.Therefore, popular machine learning models were taken into consideration to develop this project and are implemented to get the best possible results.

B. DATA PREPROCESSING

Initially, the data is explored in R by using box plots to check the outliers and the function was implemented to find missing and null values. The data types of attributes were not in the correct format, so, they were formatted in the correct order. Attributes which are supposed to be factored are converted into factor by using the function in R (as.factor(data$Attribute)), and the attributes which were to be numeric were transformed into columns by using R function (as.numeric(data$attribute)). Missing values were found and eliminated. Some columns which had many missing values, for example, Youtube view count and budget had more than 25% of missing values were removed from the master dataset because if data is imputed in place of these missing values, the observations will cause a decrease in variance and misleading classification results.

As Budget column was dropped and data set does not have the relevant information attributes for regression, the gross of a movie, regression is not practical on this data set; therefore, the proposed plan to perform regression for box office collections is dropped. The initial results of regression were not so accurate. This may be because some movies have the box office collection in seven digits and some of them in ten or more digits which caused too much variance in data and mitigating the effect of outliers.

The target variable for classification, Movie rating had some missing values or no proper rating, for example, “NOT RATED”, “NOT SPECIFIED” and “UNRATED”. These observations were removed because there is no point of classifying them as they increase the number for classification labels (>3); hence, decreasing the classification accuracy. There was only one observation for “TV-14” rating in the data set so it was removed as it can be present in only testing or training data set and will increment the count of classification labels which further decrease the accuracy.

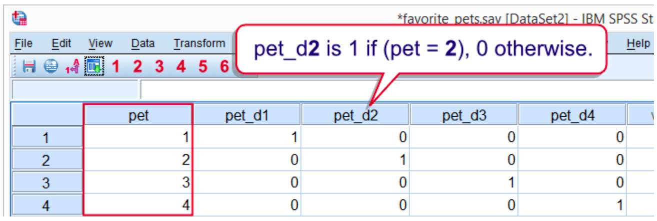
1) CREATING NEW TARGET VARIABLE FOR CLASSIFICATION

After implementing the classification on the original target variable having five classification labels, a new binary target variable **Family Movie** was built based on the description of each rating. Classification rating was divided into two groups, first group of **family movies** which has the labels (“G”, "PG" and “PG-13”) and the second group of non-Family movies (“NC-17" and “R”). The first group of movies can be watched with children whereas in other group content was restricted only for adults and people under 18 can watch only in the presence of their parent /guardians. Family Movies were labelled as “yes” and non-Family Movie’s labelled as “no”.

2) CREATING A DUMMY DATA SET AND NORMALISATION

Data must be normalised to implement PCA. However, some of the attributes are factors which cannot be used for normalisation and PCA. PCA uses Euclidian distance between the values of attributes to find the principal component; hence, categorical data is not suitable for that. Firstly, all factors attribute “Country”, “genre” and “year” were converted into dummy variables, and then normalisation is executed to implement PCA. Dummy variables are the variables where each label is divided into a single column, and that column has values 0 and 1, where 1 represents the presence of the label in a particular row and 0 for the rest of the observations, where the label is not present. This process is repeated for each label, and dummy variables were created for implementation. In total, 74 new columns were created in the process of generating dummy variable.

Figure 2: Example of Dummy Variables



pet_d**2** is 1 if (pet = **2**), 0 otherwise.

	pet	pet_d1	pet_d2	pet_d3	pet_d4
1	1	1	0	0	0
2	2	0	1	0	0
3	3	0	0	1	0
4	4	0	0	0	1

In figure 2, we can see how labels of columns “pet” are converted into dummy variables. Here, “pet_d1” column represent label “1” in “pet” column.As you can see, in the column, “pet_d1”, 1 represents the presence of label “1” in parent column “pet” and “0” used where the label “1” is not present in the column“pet”. This process was repeated for each label of “pet” column. The same process was replicated while creating dummy variables in this project. These columns generated in the process from “pet_d1” to “pet_d4” are known as **dummy variables**. The function "dummyVars(~., data)" in RStudio is used to convert the categorical columns into dummy variables. These dummy variables are suitable for normalization which were further used for implementation Principle component analysis.

C. *FEATURE SELECTION*

1) INFORMATION BY GAIN

Information by gain is used for calculating feature importance of independent variables in Movie Industry dataset by using R library "FSelector" using the function "information.gain(target variable~. , Dataset)".With the help of the library, eight important features were selected. Classification is implemented; however, improved accuracy was not observed as the dataset has only sixteen features which are already seen as a low dimensional dataset. However, improvement in computation speed is observed which is one of the significant objectives of feature selection.

2) PCA

Principal component Analysis reduces multiple attributes into one component which shows the maximum variance of those attributes. After converting original data into dummy variables, PCA is implemented, and reduced principal components are used for classification. However, due to the conversion of the original dataset into a dummy data set, an additional 74 columns were generated by converting the labels into dummy variables. A different number of components were selected according to the proportion of variance they showed, and in first implementation nine components were used for classification, however, no improvement in the accuracy.Hence, 16 componenets were used for classification but still no improvement in the accuracy. After using 16 components, further experimentation is not practical because original dataset contains 16 attributes and if more than 16 principal components are used, the objective of dimension reduction will not be achieved. **Figure 3** shows the different principle components computed for dummy dataset.

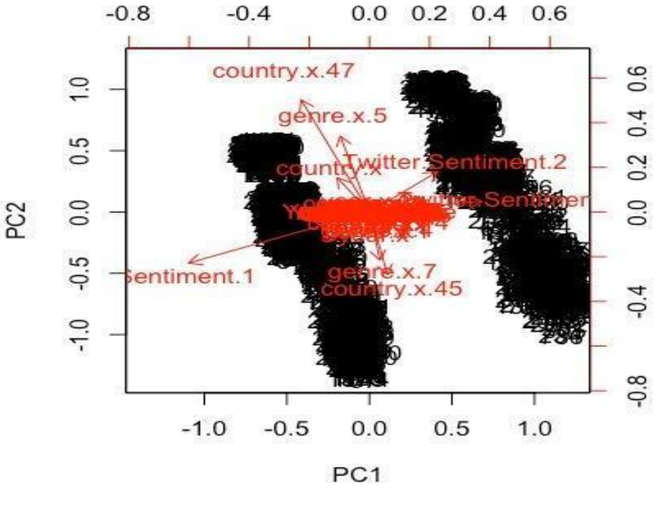


Figure 3: Principal Components

D. *CLASSIFICATION*

For classification, various types of splitting techniques, for instance, the holdout method and K-cross validation were used. Since the dataset has both numerical and factor type attributes, many classification-based machine learning algorithms were implemented to get the best results possible, and it was ensured that no information is left unanalyzed.

1) NAÏVE BAYES

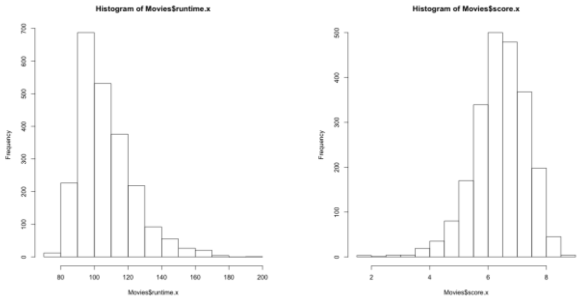


Figure 4: Histograms for Naive Bayes

Naive Bayes classifier cannot handle numerical attributes. So, first histogram was created for numerical attributes to find that if there is any possibility to create bins. In the process, two attributes were converted into bins "Runtime"(labels=c("V1","V2HIGH",V3MEDIUM","V4LOW","V5","V6","V7")) and "score" (labels=c("S1", "S2MEDIUM", "S3HIGH")). Other variables with numerical values, such as "votes"," YouTube dislike", "gross" and three other attributes were checked, but binning was not possible in them, so they dropped from the dataset for implementation of naive Bayes. Therefore, the library "e1071" in RStudio is used to implement Naïve Bayes.

2) C5.0, RANDOM FOREST AND DECISION TREE CLASSIFIERS

To concise the length of this paper, three-decision tree-based classifiers are explained here in one section as conceptually they are the same as they use decision tree algorithm for classification but with different parameter tweaking. The "C5.0", "Random Forest" and "Decision tree" classifiers were implemented after cleaning the data during the data preprocessing stage; therefore, no further processing of data is required before applying these models. Decision tree-based classifiers do not compute any distance measured between different attribute; hence, normalization is not required. The classifier can handle both numerical and factors attributes. For the implementation of random forest, one additional step is implemented which reduction of out of bag error. This step was not required for other two classifiers as out of bag error is caused by bootstrap data splitting happened inside the random forest classifier which causes out bag error [15]. The random forest classifier randomly selects the independent variables for classification whereas the decision tree classifier finds the most informative variables again and again at a causing information bias because of which all the attributes are not analysed. C5.0 is used because it has an inbuilt pruning of tree which reduces the overfitting. C5.0 uses information gain for implementation of decision tree classifier. Once the finding of parameters with the least amount of out-of-bag error is done, classification is implemented resulting in improvement in both, the classification and computation speed. The best results were achieved at the number of trees = 3, node = 5 and sample size = 1351 for the random forest.

E. *ARTIFICIAL NEURAL NETWORK*

This is a black box method used to classify the target variable and, in an attempt, to achieve higher accuracy. Multiple tweaking of the parameters of *H2o.deeplearning* functions were experimented to implement ANN. For example, the grid search function is utilized to loop various implementation of ANN on the training data set and validated on small validation dataset extracted from the testing dataset using the "h2o.getGrid". In grid function, number of hidden layers were changed with a different approach, for example, narrower hidden layers where number of neutrons were reduced at every next hidden layer.Also number of backpropagation was changed which is the number of the epoch. Additionally, different activation layers ("Tanh", "RectifierWithDropout", "Maxout") were tried with different regularisation parameter "l1". The H2o deep learning function requires a data in the form of H2o data frame,therefore training and testing datasets were converted as h20 data frame by using the function "as.h20(data set)".

F. *SUPPORT VECTOR MACHINE*

Support Vector machine is implemented after initial data preprocessing without any changes in the data set. Initially "kernlab" library is used to implement "ksvm()" function with kernel = linear but classification accuracy achieved was low. Therefore, the "e1071" library was used and a "tune()" function is used to find the best SVM parameters. The best parameters founded were "SVM-Kernel= Radial", "cost=4", "gamma=0.0625" and "Number of Support Vectors:1347". These parameters are used for classification and improvement in classification accuracy is observed.

G. *BOOSTING ALGORITHMS*

1) *ADA-BOOST*

The ADAboost classifier was implemented with the decision tree as a base classifier. The ADAboost classifier is based on a boosting algorithm in which a weak learner is used and optimised to do complex classifications. It is an ensemble approach where multiple weak classifiers are implemented, and every time a new classifier is implemented, the misclassified labels from the previous classifier are corrected by adding the weights to them in the next classification. The "fastboost" library is used to implement "ADAboost()" classifier with 1000 iterations. This model is implemented on the family movie target variable which has the binary labels "yes" and "no". The improved result with accuracy above 70% is achieved.

2) *GRADIENT BOOSTING*

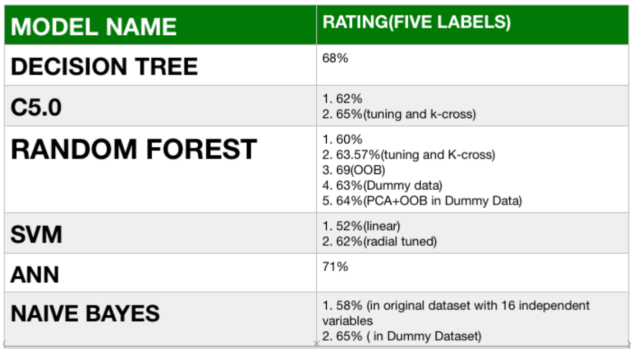
The gradient is another boosting algorithm like ADAboost which uses a weak classifier to build a complex model. However, in this boosting approach boost error rate is checked in every classification, and this algorithm tries to reduce the error rate from the previous classification and builds a robust and complex classifier. Library "gbm" is used and "train ()" function is used to implement this classifier. 10 K-fold cross-validation is used, and tune length = 3 is used in classifying the target variable, Family Movie. Classification accuracy of more than 70% is achieved with the help of the model.

The next section will evaluate and discuss the results achieved by the models mentioned above in detail.

VI. EVALUATION AND RESULTS

The classification results obtained by the implementation is shown in a table below. As classification is implemented on variable created based on MPAA rating, "Family Movie", the classification results for this target variable are shown in Table 5. The Table 4 shows the classification results of the target variable MPAA rating with five labels. Table 6 shows the classification results for "Family Movie".

Table 4: Classification of MPAA rating- 5 labels



MODEL NAME	RATING(FIVE LABELS)
DECISION TREE	68%
C5.0	1. 62% 2. 65%(tuning and k-cross)
RANDOM FOREST	1. 60% 2. 63.57%(tuning and K-cross) 3. 69(OOB) 4. 63%(Dummy data) 5. 64%(PCA+OOB in Dummy Data)
SVM	1. 52%(linear) 2. 62%(radial tuned)
ANN	71%
NAIVE BAYES	1. 58% (in original dataset with 16 independent variables 2. 65% (in Dummy Dataset)

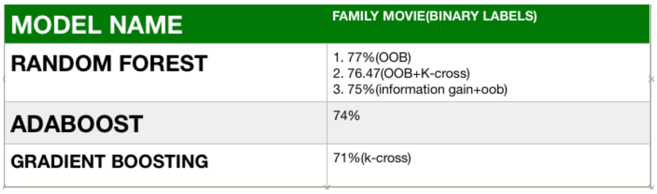
Evaluation of results obtained by classifying MPAA rating:

Various classification models are implemented to classify this target variable, but maximum classification accuracy was obtained by ANN (Artificial Neural Network), a deep learning algorithm. The second highest accuracy was achieved by random forest (see table 5) after tweaking its parameters such that minimum out-of-bag error occurs. In table 5, the random forest's first two results show that after using K-cross validation, accuracy is improved by 3% in comparison to Hold-out method. The exact same can be observed for "C5.0" classifier. In Decision Tree classifier, 68% classification Accuracy is achieved.

In Support Vector machine, significant improvement was observed when the kernel is changed.An increase of 10% is see once the kernel in SVM is changed from "linear" to "radial".

On the other hand, Naïve Bayes portrayed interesting results while implementing. Classification accuracy is better when Naive Bayes was implemented on Dummy data (65%) with an increase in the accuracy of 7% when compared with the accuracy of Naïve bayes on original dataset (58%). The same pattern was observed for Random Forest where without implementing K-cross validation, a better accuracy (63.57%) is observed in comparison to similar implementation on the master dataset without dummy variables. In table 5, result 1 and 4 of Random Forest show the comparison between the results achieved. In random forest, by using PCA and k-cross on Dummy data, minor improvement in results is observed than the result obtained by master data set with K-cross validation (see table 4: Random Forest and compare results 2 and 5).

Table 5: Binary Classification Results



MODEL NAME	FAMILY MOVIE(BINARY LABELS)
RANDOM FOREST	1. 77%(OOB) 2. 76.47(OOB+K-cross) 3. 75%(information gain+oob)
ADABOOST	74%
GRADIENT BOOSTING	71%(k-cross)

The objective of creating this target variable is to get better classification accuracy and to group the movies as a family movie or not. The creators of the movie would be able to know in which category their movie falls which makes a massive difference in terms of business of the movie. The intention behind this target variable was achieved as highest classification accuracy of 77% is achieved in this project. Firstly, random forest model was optimized for least Out-Of-Bag error and then classification is implemented. Other ensemble classification models, for example random forest were implemented but with boosting techniques. ADABOOST and Gradient Boosting classifiers achieved the classification accuracy of 74% and 71% respectively. ADABOOST performed better than Gradient boosting with 3% more classification accuracy than Gradient boosting.

FINDINGS

Firstly, after exploring the target variable, MPAA ratings, it is found that the target variable is unbalanced. There are a total 5 labels "R", "NC-17", "PG", "PG-13" and "G". As most of the observations belong to "PG" and "R", 70% of the target variable is unbalanced, because of which K-cross validation is used to train and test the all the observations which is not possible in Hold-Out method and type-I error is more than type-II error.

CONCLUSION

A classification of MPAA rating was implemented successfully and additionally a new target variable is created to group the movie as a family movie based on MPAA rating(s). Eight classification-based machine learning algorithms were implemented successfully with proper evaluation by K-cross validation and feature selection by information gain during this project. Further dimension reduction technique, PCA was implemented for which data was normalized by creating dummy variables. Also, Out-Of-Bag error was reduced for the random forest algorithm before implementing. Highest classification accuracy of 71% was achieved by ANN for classifying MPAA rating and 77% was achieved by Random forest for classifying Movies as a family movie or not.

In future, there is a potential of improving the classification accuracy of MPAA rating by 29% with addition of more data that contains more information regarding target variable, which can be the data related to the movie's crime scenes, romantic scenes, comedy scenes etc. which will have more information gain in predicting MPAA ratings. Another future scope of this project is classifying a movie's MPAA rating by analyzing the script of the movie using NLP which will reduce human intervention.

REFERENCES

[1] A. Oghina, M. Breuss, M. Tsagkias, and M. De Rijke, "Predicting imdb movie ratings using social media," presented at the European Conference on Information Retrieval, 2012, pp. 503–507.

[2] "Movie Industry | Kaggle." [Online]. Available: https://www.kaggle.com/danielgrijalvas/movies. [Accessed: 18-Apr-2019].

[3] X. Ning, L. Yac, X. Wang, B. Benatallah, M. Dong, and S. Zhang, "Rating prediction via generative convolutional neural networks based regression," *Pattern Recognition Letters*, 2018.

[4] M. H. Latif and H. Afzal, "Prediction of movies popularity using machine learning techniques," *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 16, no. 8, p. 127, 2016.

[5] J. Han, J. Pei, and M. Kamber, *Data mining: concepts and techniques*. Elsevier, 2011.

[6] S. Bashir, U. Qamar, F. H. Khan, and M. Y. Javed, "An efficient rule-based classification of Diabetes using ID3, C4. 5, & CART ensembles," presented at the 2014 12th International Conference on Frontiers of Information Technology, 2014, pp. 226–231.

[7] D. Delen and R. Sharda, "Predicting the financial success of hollywood movies using an information fusion approach," *Indus Eng J*, vol. 21, no. 1, pp. 30–37, 2010.

[8] M. H. Latif and H. Afzal, "Prediction of movies popularity using machine learning techniques," *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 16, no. 8, p. 127, 2016.

[9] V. Nithin, M. Pranav, and P. Sarath Babu, "Lijiya: A predicting movie success based on IMDB data," *Int. J. Data Min. Tech*, vol. 3, pp. 365–368, 2014.

[10] E. Muñoz, S. Yumusak, E. Dogdu, P. Minervini, and H. Kodaz, "A hybrid method for rating prediction using linked data features and text reviews," *Know@LOD*, 2016.

[11] M. Riegler *et al.*, "Right inflight?: A dataset for exploring the automatic prediction of movies suitable for a watching situation," presented at the Proceedings of the 7th International Conference on Multimedia Systems, 2016, p. 45.

[12] X. Geng and P. Hou, "Pre-release prediction of crowd opinion on movies by label distribution learning," presented at the Twenty-Fourth International Joint Conference on Artificial Intelligence, 2015.

[13] M. S. Rahim, A. Z. M. E. Chowdhury, M. A. Islam, and M. R. Islam, "Mining trailers data from youtube for predicting gross income of movies," in *2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*, 2017, pp. 551–554.

[14] "Motion Picture Association of America," *Wikipedia*. 08-Apr-2019.

[15] S. Janitza and R. Hornung, "On the overestimation of random forest's out-of-bag error," *PloS one*, vol. 13, no. 8, p. e0201904, 2018.