Movie Rating Prediction using Ensemble Learning Algorithms

Zahabiya Mhowwala[1], A. Razia Sulthana[2], Sujala D. Shetty[3]
Department of Computer Science
Birla Institute of Technology and Science
Dubai, United Arab Emirates

***Abstract*—Over the last few decades, social media platforms have gained a lot of popularity. People of all ages, gender, and areas of life have their presence on at least one of the social platforms. The data that is generated on these platforms has been and is being used for better recommendations, marketing activities, forecasting, and predictions. Considering predictions, the movie industry worldwide produces a large number of movies per year. The success of these movies depends on various factors like budget, director, actor, etc. However, it has become a trend to predict the rating of the movie based on the data collected from social media related to the movie. This will help a number of businesses relying on the movie industry in making promotional and marketing decisions. In this report, the aim is to collect movie data from IMDB and its social media data from YouTube and Wikipedia and compare the performance of two machine learning algorithms – Random Forest and XGBoost – best known for their high accuracy with small datasets, but large feature set. The collection of data is done from multiple sources or APIs.**

***Keywords*—Machine learning; ensemble learning; random forest algorithm; XGBoost; movie rating prediction**

I. Introduction

Living in a socially and digitally connected world, everyone leaves a digital trace of themselves in different forms on the web. People are actively sharing their emotions, feelings, opinions and views through social communications. These communications generate a huge amount of data which is being analyzed to predict, recommend, monitor or cope with varied events, from simple matters to complex problems.

The movie success prediction has a vast range of attributes that gives a holistic approach to perform predictions, movies are something that creates lots of buzz in digital space, and based on the stardom of celebrities there are lots of hailing and criticism bubbling on social media platforms. Also, movie prerelease phase is very strategic and well defined which includes releasing teasers, trailers, a series of movie promotion activities, etc. Connecting the dots, these dots are the number of views, page counts, and sentiments of people on social media. But everyone's views and comments do not hold equal values. There are influencers and well-known critics whose views and comments hold more value than rest. Many social media platforms by default work on an algorithm where they rank the comments based on popularity and relevance thus, rather than collecting all of the comments of each video, only the comments which hold greater relevance and enough values to cover almost every point of view were collected. This strategy allowed to trace the digital footprints of buzz and move closer to predict the future of the product.

In everyday lives, when people have to make important decisions it is highly unlikely that the decision is based on only one factor. Rather, the decision is always based on the collective opinions coming from a lot of different sources. Ensemble learners operate on a very similar idea, where decisions from multiple learning models are combined to improve the performance of the output. The idea is to combine the results of weak individual models which are prone to overfitting and generate a model which would reduce the error. The simple techniques used for combining the results include – Maximum Voting, Averaging, and Weighted Averaging. Some advanced techniques include – Model Stacking, Blending, Bagging and Boosting.

Random forests are bagging ensemble method that use decision trees as weak learners. It performs feature bagging along with randomly selecting training data for each of its learner models. The random forest combines hundreds or thousands of decision trees, trains each one on a slightly different set of the observations, splitting nodes in each tree considering a limited number of features. The final predictions of the random forest are made by averaging the predictions of each individual tree. This reduces the correlation between the different predictions and makes the process more robust to missing values.

Gradient boosting is a boosting ensemble that trains many models in a gradual, additive, and sequential manner. It does so by minimizing the loss function, by repetitively leveraging the residuals (or error) of the previous model and strengthening it. XGBoost is the optimized gradient boosting algorithm using parallelization for tree building, tree pruning and regularization to avoid overfitting, efficient handling of missing data and hardware optimization. There are many hyperparameters that needs to be set for efficient use of this algorithm, it includes – n_estimaters, max_depth, eta or learning rate, reg_alpha & reg_lamba (regularization terms).

A. Problem Statement

Prior knowledge about the success or failure of a particular movie and what factors affect the success will benefit the production houses in how to go about with promoting and other expensive business decisions. For example, if a movie does well in test screenings or if they anticipate good reviews [1] from social media due to promotional activities, they can decide to release it on an opening weekend in more theaters in

hopes of bringing in more revenue. One of the ways the success of a movie is identified is through its ratings, predicting the ratings of a movie before it is released based on the data available at that moment will help in decision making.

The objective is to predict the ratings of the movies using two ensemble learning algorithms - Random Forest and XGBoost, that can be used to evaluate the success or failure of a movie before its release and then to compare them on their performance. The data will be collected from IMDb, people's reviews on the movies will be collected from YouTube comments, and other information to strengthen the model will be collected from Wikipedia page of the movie. The rest of the paper is organized in following fashion. Section 2 discuss the literature work related to the proposed work. Section 3 details the data collection procedure. Section 4 describes the proposed model and the implementation procedure. Section 5 explains the evaluation results. Section 6 concludes the research work.

II. Literature Review

There have been many different models proposed for prediction of movie ratings. Different papers included different types of data related to the movie to evaluate its outcome on ratings. Yasseri et al. [2] proposes a linear regression model for predicting the box office revenue. The model works well for successful movies, but for not so successful movies the model fails to provide accurate predictions. The movie data set used for this model is very small. The data used for the model [3] is from IMDb movie data and social media data from Twitter and Wikipedia. These features are then used in factorization machines to predict the ratings of the movie. The model gave 0.88 R^2 score. In this study [4], the features are classified into Who, What, and When. A lot of different binary and multi-class classification algorithms are used and all of them are analyzed based on ROI, accuracy, precision, recall and the best is selected.

Dhir and Raj [5], compares several machine learning algorithms - SVM, Random Forest, Ada Boost, Gradient Boost and KNN on data from IMDb. Random Forest gave the best accuracy (83%) in terms of success prediction. Predicting if the movie is a hit or flop [6] using 3 features– IMDb ratings, MusicRatings, and No of Screens. Logistic Regression algorithm is used. The model has achieved an accuracy of 80%.

Magdum and Megha uses sentiment analysis [7] using S-PLSA, and also used ARSA (Autoregressive Sentiment Analysis) model for predicting the sales performance, on the reviews and tweets. The paper [8] proposes to integrate movie data from IMDb with the user response from social media on the movie. This will give more features that can help in making better predictions. This paper [9] tries to find the relation between blogs and movie and music success. The paper could predict the monetary success with 79.84% precision value using Decision Tables. It is vague about how the features were selected from the blogs and how it's used in the algorithms. Santosh et al. [10] aimed in analyzing the effect of hype of the movie on the success of the movie. The hype factor was calculated from number of tweets per second related to the movie, on Twitter. The tweets were taken seven days before the release date of the movie. Real-time sentiment and Twitter data analysis [17], [18] is used in various applications.

This paper [11] has illustrated the use of different regression models, neural networks and classification algorithms to predict the success of Bollywood movies using data from Wikipedia, RadioMirchi and BoxOfficeIndia. The data is normalized before feeding into models. This study [12] not only predicts movie grosses before the release, but also for newly released movies. It says that some of the movies do not do well in the first week but later on it does become successful due to publicity, Oscar nominations and other factors.

Some other studies included prediction of housing prices [13], flight delays [14], crowdsourcing quality [15], etc. using ensemble learning models. For housing price prediction, the paper collected the data from Pune region, extracted the relevant features and used CART Decision Tree and Random Forest Regressor models to perform the analysis. In all of the tests, the RF performed better. Prediction of delay in flights was performed on US flights data after the analysis of features, using Gradient Boosted Decision Trees. Another paper predicted and controlled the quality of crowdsourcing data using Random Forests algorithm, thereby improving the ability and efficiency of the contractor in identifying deceptive participants and take necessary actions.

All of these earlier studies, as mentioned, have tried to either identify the factors that affect the ratings or concentrated on finding a better model for prediction. The studies before on rating prediction had not taken the sentiments of the audience into consideration, before the release of the movie. This study aims at collecting and generating a dataset from different sources that includes the movie's data and social media related data (from YouTube and Wikipedia) before the movie's release and then using this dataset in ensemble learning algorithms to perform predictions of the movie's rating and to compare the performance of the algorithms. Also, this study aims at analyzing the effect of social media data on the predictions. All of these accumulated together is handled by a separate stream of machine learning called Recommendation System (RS) [16].

III. Data Collection Procedure

The most important and basic step for training a model is using an enriched dataset. In this case, it does not exist as such and had to be collected; Our primary objective is to get the data from different platforms. For this, we have used data from IMDB, Wikipedia, and YouTube.

- Movie related metadata: Sources like TMDb and OMDb which expose their APIs to retrieve IMDB movie data were used. To start with these APIs, IMDb ids of the movies had to collected. To get the list of the ids of the movies released in late 2018 or later, an IMDb list of movies was scraped using a script. These ids were then used to get movie related data such as directors, writers, cast, runtime, votes, ratings, etc.
- Wikipedia data: The popularity of movies was collected through Wikimedia REST API for Wikipedia page views of the movie – 30 days prior to the release date of the movie. To retrieve page views from the API, it

required Wikipedia links of the form "Parasite (2019_film)" for the movie Parasite. So these links were first scraped using Google search and then used with the API.

- YouTube data: To understand the buzz created, data from YouTube's API was collected to analyze likes, shares, and comments of the official trailer and teaser. Again, the API needed the video id to get any sort of information on the videos. The YouTube video ids of the trailers and teasers of the movies, which are uploaded before the movie is released, were scraped using Python scripts. And then the comments and other video statistics were collected through API.

Data to be analyzed is collected in the range of – movies released between late 2018 and 2019. The focus is on the movies that are relatively new as the trend and popularity of movies from older times is irrelevant from the trend today.

A. *Data Pre-Processing and Transformation/Feature Selection*

All the data retrieved from different sources and platforms can be termed as raw data as it lacks the uniformity and contains missing value, hence a wrapper script was prepared that would make a uniform data to include relevant data and discard the rest. Drilling down more into data obtained from each source and their relevance, scripts returning values from TMDb and OMDb are combined to obtain movie metadata which includes attributes such as genre, runtime, directors, actors, budget, etc. And social media data from Wikipedia and YouTube gives us attributes such as page view count, likes, dislikes, comment count, comments, etc. The movie metadata features are given in Table I and social media data is given in Table II.

The comments that were collected were of the trailers and teasers of the movie. Comments from social media cannot be used as such; hence we calculated the sentiment of comments to analyze its impact on the rating of the movie in the future, using Cognitive Services from the Microsoft Azure platform. The sentiment is calculated and categorized into positive, negative and neutral, where each category is given a score from 0 to 1 depending on the sentiment of the comment. We will be dealing with data of 4045 movies with following features.

Additional extracted features (feature selection), mentioned in Table III, were calculated from the data – categorical data was converted to numerical columns using one hot encoding, the popularity of the directors, writers, cast, and production companies among the dataset was calculated by taking the mean of the ratings of the movies they were involved in (using target/mean encoding), and since the release date had large number of values when compared to the number of records, only the month was taken into consideration.

Fig. 1 and Fig. 2 shows the interaction of movie ratings (imdbRating) with the extracted features – mean rating of the director and the ratings of production companies. The correlation matrix of the dataset which gives how each feature is related with other features is given in Fig. 3. Positive values denote higher correlation.

TABLE I. MOVIE METADATA

Feature Name
Genre
Runtime
Rated
Production Companies
Vote Count
Rating
Adult

TABLE II. SOCIAL MEDIA DATA

Feature Name
Wikipedia View Count
YouTube video comment count
Video Likes
Video dislikes
Video Comment's sentiment
Video view count

TABLE III. EXTRACTED FEATURES

Feature Name	**Description**
Directors Movie Sum	Average of total movie number made by the directors
Directors Rating Mean	Average of arithmetic means of the ratings of the all movies made by the director
Cast Movie Sum	Average of the total movie numbers in which the first three leading cast played
Cast Rating Mean	Average of the arithmetic means of the ratings of the all movies in which the cast played
Writers Movie Sum	Average of total movie number
Writers Rating Mean	Average of arithmetic means of the ratings of all the movies
Production Company Sum	Average of total number of movies produced by the companies
Production Company Rating Mean	Average of the arithmetic means of the ratings of all the movies produced by each company
Release Month	Release month of the movie extracted from release date

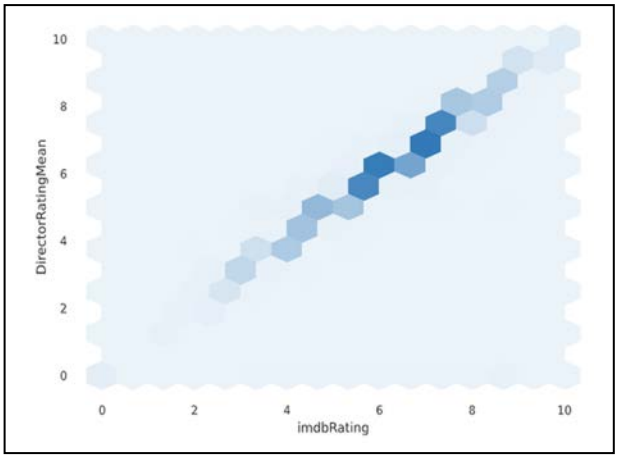

Fig. 1. Interaction of Director Mean Rating with imdbRating.

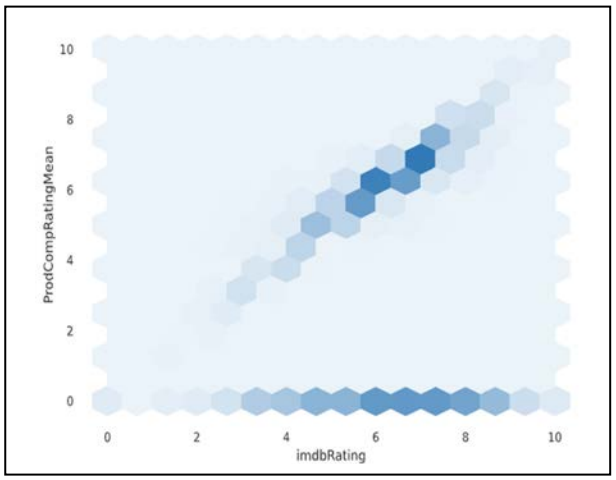

Fig. 2. Interaction of Production Company Ratings with imdb Ratings.

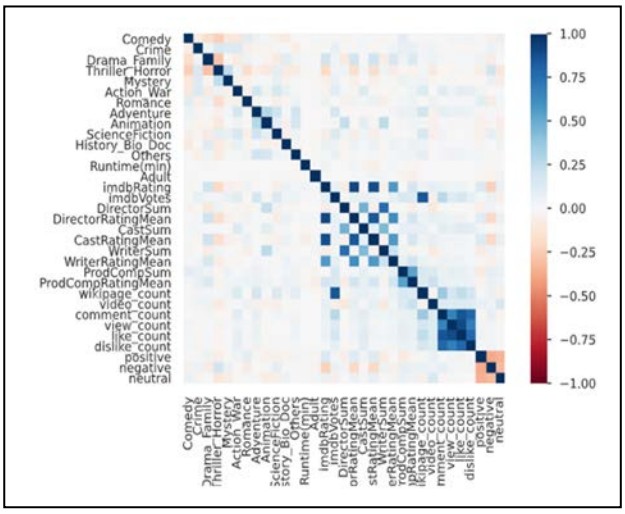

Fig. 3. Correlation Matrix.

IV. Model Implementation using Scikit's Learn Library

In order to explore how social media data affects the prediction accuracy, two datasets were made based on their sources. One was with movie metadata and social media data and the other with only movie metadata without social media data from Table II. Then, random forest and xgboost were implemented using both these sets of data.

After getting the data ready, implementing the algorithm in Python is done using an efficient and widely used package called Scikit-learn. It has provided methods to perform classification and regression tasks. We will be using RandomForestRegressor and XGBRegressor from xgboost class.

A parameter is a configuration variable that is internal to the model and it is very important for finetuning the model. It is also responsible for overall performance of the model and its value can be estimated from the given data. Both of these algorithms have hyperparameters that when set properly, avoids overfitting. Some of these parameters that we have used are given below.

- RandomForestRegressor – n_estimators, max_depth, min_samples_split, min_samples_leaf, random_state.
- XGBRegressor – n_estimators, learning rate, objective, reg_lambda, reg_alpha.

These parameters define how many trees are built for training the model and different attributes for the tree. Also, it defines how many samples are to be picked during the building of a tree (bootstrapping). The learning rate and regularization terms to control overfitting are some of the other parameters.

V. Testing and Evaluation Results

To calculate the performance metrics, 10-fold cross validation was performed using both the algorithms on the training data that was randomly selected from datasets. The performance score was averaged over all the scores for each model and datasets. Different metrics are displayed in the Table IV and Table V.

In a situation where dataset is small, and there is a possibility that the model might overfit the training subset of the data, cross validation uses the whole dataset for training and testing and hence avoiding overfitting to some extent. It also reassures that the model will work well for unseen data.

We also calculated the mean squared error (1); root mean squared error (2) and mean absolute error (3) for each of the models (shown in the Table IV). These functions consider equally - the negative and positive deviation from the target value. But, mean squared error (MSE) penalizes the predicted errors which are further away from the target value more. Whereas, the mean absolute error (MAE) does not penalize based on this deviation. That is, MSE is sensitive to outliers but MAE is robust to outliers.

$$\frac{1}{n}\sum_{t=1}^{n} e_t^2 \quad (1)$$

$$\sqrt{\frac{1}{n}\sum_{t=1}^{n} e_t^2} \quad (2)$$

$$\frac{1}{n}\sum_{t=1}^{n} |e_t| \quad (3)$$

TABLE IV. Performance Metrics (Movie Metadata + Extracted Features)

Performance Metric	Random Forest	XGBoost
Cross Validation Score	0.894	0.931
MSE	0.28	0.18
RMSE	0.53	0.42
MAE	0.37	0.22

TABLE V. Performance Metrics (Movie Metadata + Extracted Features + Social Media Data)

Performance Metric	Random Forest	XGBoost
Cross Validation Score	0.907	0.953
MSE	0.16	0.09
RMSE	0.40	0.30
MAE	0.25	0.13

Fig. 4 and Fig. 5 are the plots of predicted values to target values for dataset with social media data. For comprehensibility, only 200 points are displayed in the plots. The red line shows the predicted values and the blue dots display the original or target values. From the plots, it is easily understood that xgboost predicts more accurately than random forest.

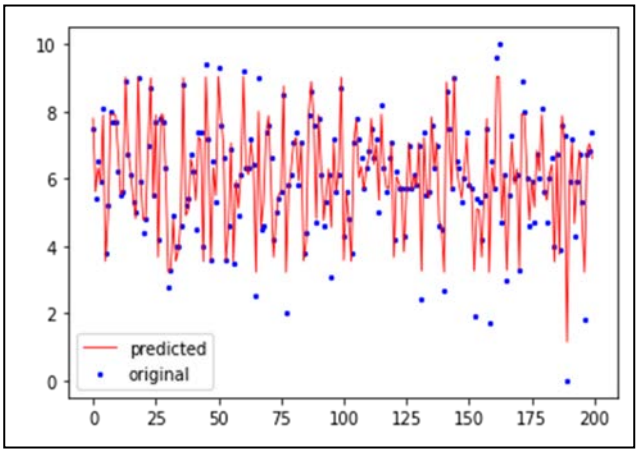

Fig. 4. Target (Original) vs Predicted Plot for Random Forest.

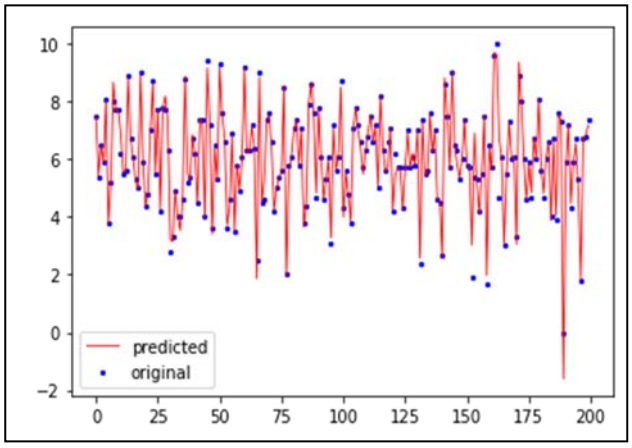

Fig. 5. Target (Original) vs Predicted Plot for XGBoost.

The implementation results show that XGBoost is the better performer than the Random Forest algorithm for both datasets. And the data that contains both metadata and social media data performs better in predicting the ratings. Though, Random Forest in itself gives a good performance measure, it works well on training and test data.

VI. Conclusion and Future Scope

In this paper, comparison of two ensemble learners to predict a continuous output was carried out. It was observed that xgboost being a more efficient algorithm performed better when compared to random forest. But the cross-validation score which is 0.95 for xgboost, could lead to overfitting if not properly regularized. There are several hyperparameters that needs to be set to achieve the same. Used movie's metadata with and without its social media data to see how it affected the predictions. From the correlation matrix, it is seen that the popularity of directors, actors and writers affect the ratings most.

For the future work, this can be extended by including a greater number of movies. Since the dataset needed the Wikipedia page counts and YouTube comment sentiments before the movie was released, the dataset in this paper included the movies from only 2019-2020. The data from before this is scarce and would need more preprocessing, if considered.

Further study can also include data from other social media platforms such as Twitter, or blog related features of the movies. This will give the hype being created around the movie and the effect of this can be analyzed on the ratings. Twitter platform gives the popularity of people connected with the movie and how their popularity and social media activity might lead to more people going out to watch their movies and eventually rating it. For example, in this paper the director rating means and cast rating means were extracted from the dataset, these ratings give the popularity of the people amongst the audience based on the movie ratings. The popularity of the cast and director affects the ratings of the movie, as seen from the interaction diagrams discussed. The popularity calculated through twitter followings and retweets can be considered and analyzed for their effect on prediction of movie ratings. This study can also be extended to predict the revenue that would be generated, before a movie is released. The factors that affect the revenue generation will have to be considered for prediction.

References

[1] A. R. Sulthana and S. Ramasamy, "Context based classification of Reviews using association rule mining, fuzzy logics and ontology," Bulletin of Electrical Engineering and Informatics 6, no. 3, 250-255, 2017.

[2] T. Y.,. J. K. Márton Mestyán, "Early Prediction of Movie Box Office Success Based on Wikipedia Activity Big Data," PLoS ONE, vol. 8, no. 0071226, 2013.

[3] S. G. O. Beyza Çizmeci, "Predicting IMDb Ratings of Pre-release Movies with Factorization Machines Using Social Media," in (UBMK '18) 3rd International Conference on Computer Science and Engineering, Turkey, 2018.

[4] K. Z. Michael Lash, "Early Predictions of Movie Success: the Who, What, and When of Profitability," Journal of Management Information Systems, vol. 33, 2017.

[5] A. R. Rijul Dhir, "Movie Success Prediction using Machine Learning Algorithms and their Comparison," in International Conference on Secure Cyber Computing and Communication (ICSCCC), India, 2018.

[6] G. Verma and H. Verma, "Predicting Bollywood Movies Success Using Machine Learning Technique," in 2019 Amity International Conference on Artificial Intelligence (AICAI), Dubai, United Arab Emirates, 2019.

[7] S. S. Magdum and J. V. Megha, "Mining Online Reviews and Tweets for Predicting Sales Performance and Success of Movies," in International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2017.

[8] A. Bhave, H. Kulkarni, V. Biramane and P. Kosamkar, "Role of different factors in predicting movie success," in International Conference on Pervasive Computing (ICPC), Pune, India, 2015.

[9] F. Abel, E. Diaz-Aviles, N. Henze, D. Krause and P. Siehndel, "Analyzing the Blogosphere for Predicting the Success of Music and Movie Products," in International Conference on Advances in Social Networks Analysis and Mining, Odense, Denmark, 2010.

[10] P. K. Ajay Siva Santosh Reddy, "Box-Office Opening Prediction of Movies based on Hype Analysis through Data Mining," International Journal of Computer Applications, vol. 56, no. 1, 2012.

[11] A. Kanitkar, "Bollywood Movie Success Prediction using Machine Learning Algorithms," in 3rd International Conference on Circuits, Control, Communication and Computing (I4C), Bangalore, India, 2018.

[12] I. R. S. Jeffrey S. Simonoff, "Predicting movie grosses: Winners and losers, blockbusters and sleepers," CHANCE, vol. 13, no. 3, 2012.

[13] R. Sawant, Y. Jangid, T. Tiwari, S. Jain and A. Gupta, "Comprehensive Analysis of Housing Price Prediction in Pune Using Multi-Featured Random Forest Approach," in 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 2018.

[14] S. Manna, S. Biswas, R. Kundu, S. Rakshit, P. Gupta and S. Barman, "A statistical approach to predict flight delay using gradient boosted decision tree," in 2017 International Conference on Computational Intelligence in Data Science(ICCIDS), Chennai, India, 2017.

[15] H. Lan and Y. Pan, "A Crowdsourcing Quality Prediction Model Based on Random Forests," in 2019 IEEE/ACIS 18th International Conference on Computer and Information Science (ICIS), Beijing, China, 2019.

[16] A. R. Sulthana and S. Ramasamy, “Ontology and context based recommendation system using Neuro-Fuzzy Classification,” in Computers & Electrical Engineering, 74, 498-510, 2019.

[17] Lekha R. Nair, Sujala D. Shetty, Siddhanth D. Shetty, “Streaming Big Data Analysis for Real-Time Sentiment Based Targeted Advertising”, International Journal for Electrical and Computer Engineering, Institute of Advanced Engineering and Science , 7.1, pp. 402-407, 2017.

[18] Lekha R. Nair, Dr. Sujala D. Shetty, “Streaming Twitter Data Analysis using SPARK for effective Job Search”, Journal of Theoretical and Applied Information Technology (E_ISSN 1817-3195/ISSN 1992-8645).