International Journal of Trend in Scientific Research and Development (IJTSRD)
Volume 5 Issue 4, May-June 2021 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

Drug Review Sentiment Analysis using Boosting Algorithms

Sumit Mishra

Electrical and Electronics Department, Bharati Vidyapeeth's College of Engineering, New Delhi, India

ABSTRACT

Sentiment Analysis of the Reviews is important to understand the positive or negative effect of some process using their reviews after the experience. In the study the sentiment analysis of the reviews of drugs given by the patients after the usage using the boosting algorithms in machine learning. The Dataset used, provides patient reviews on some specific drugs along with the conditions the patient is suffering from and a 10-star patient rating reflecting the patient satisfaction. Exploratory Data Analysis is carried out to get more insight and engineer features. Preprocessing is done to get the data ready. The sentiment of the review is given according to the rating of the drugs. To classify the reviews as positive or negative three Classification models are trained LightGBM, XGBoost, and CatBoost and the feature importance is plotted. The result shows that LGBM is the best performing Boosting algorithm with an accuracy of 88.89%.

KEYWORDS: *Sentiment Analysis, NLP, Classification, textblob, Features Engineering*

How to cite this paper: Sumit Mishra "Drug Review Sentiment Analysis using Boosting Algorithms" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-5 | Issue-4, June 2021, pp.937-941, URL: www.ijtsrd.com/papers/ijtsrd42429.pdf

IJTSRD42429

I. INTRODUCTION:

Sentiment Analysis of the reviews is a very important aspect as it helps companies perceive how positively the customers are perceiving the product and to make changes according to user satisfaction and help them grow. Doing the sentiment analysis on the drug review is also an application where the drugs which don't have any effect on the condition can be changed with something effective and the side effect of the drug can also be analyzed.

The dataset used, contains the reviews of the drugs given by the patients according to their experience after the usage of the drug and a rating from 1 to 10 is given for the specific drug [8]. The purpose of this study is to do the sentiment analysis on the drug reviews given by the patients using the gradient boosting algorithms in machine learning.

Initially, the dataset was segregated into two parts which are train and test set but as the main task is to do the sentiment analysis so both the dataset is merged to get more data to train and test overall (In this study we'll talk about the merged dataset only). Exploratory Data Analysis is done on the dataset with different features to gain insight about the data and the correlation between them which will help in feature engineering. The features like 'uniqueID' are not of much use as they are just the identity given to each of the data point or patient review. some preprocessing is done before the EDA so the data is ready for the Exploratory Data Analysis. Different bar graphs, Histogram and Word Clouds are plotted. The Reviews then are cleaned so that unnecessary words and elements are removed and the features are generated. Feature Engineering is done on both the uncleaned and cleaned reviews. Textblob module is also used to give the polarity to both the cleaned and uncleaned reviews and use it as a feature as well. The classification models are trained and their performances are compared on the basis of the evaluation metrics given in the study. Accuracy score is sometimes misleading when the dataset is skewed so metrics like recall and precision are also given for deeper analysis. The Classification algorithms that are used are LightGBM, XGBoost, and CatBoost.

Previous research which are related to the drug review sentiment analysis were carried out by Jin-Cheon Na and Wai Yan Min Kyaing where they did the sentiment analysis with three values positive, negative and neutral and implement the SVM [10]. In the early studies like Pang et al [11] which was focused on document level analysis for the sentiment assigning. Some recent researcher like Jo & Oh [12] and Ding et al [13] carried out the sentence level sentiment analysis to examine the opinion or review. Some researcher like Liu [14] have used linguistic features in addition to the word features to overcome the limitation of the Bag-of-Word approach. In this study, I used the gradient boosting algorithms in machine learning for the sentiment analysis. The results of the three models are also compared based on the evaluation metrics.

II. Dataset

The Dataset is taken from the UCI Machine Learning Repository [7]. The dataset contains the reviews of the drugs given by the patients according to their experience after the usage of the drug and a rating from 1 to 10 is given for the specific drug [8]. The Drug Review Data Set is of shape (215063, 7) i.e. It has 7 features including the review and 215,063 Data Points or entries. The features are 'drugName' which is the name of the drug, 'condition' which is the condition the patient is suffering from, 'review' is the patients review, 'rating' is the 10-star patient rating for the drug, 'date' is the date of the entry and the 'usefulcount' is the number of users who found the review useful. The sentiment of the Drug review is the target variable that the models will train to predict. Here we can notice that the sentiment of any review is not given, so we have to give the sentiment to the rating first and then use it as the target variable. The drugName and condition are categorical features, the date is date object, rating and usefulcount are numerical features, and the review is text. These reviews are from the year 2008 to 2017.

III. Exploratory Data Analysis

Exploratory Data Analysis (EDA) is done to get an insight into the data and summarize the main characteristics. To understand dependency or correlation of the features. The plots are generated using the matplotlib [5] and seaborn [6] library.

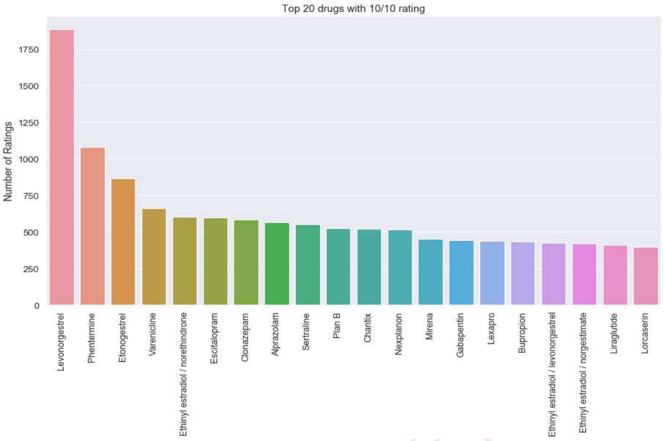

Fig 1 Top 20 drugs with 10/10 ratings

Figure 1 is a bar graph which shows the top 20 drugs given in the data set with a rating of 10/10. 'Levonorgestrel' is the drug with the highest number of 10/10 ratings, about 1883 Ratings in the data set for 'Levonorgestrel'. It's followed by 'Phentermine' with 1079 ratings.

Levonorgestrel (LNG) is a synthetic progestogen similar to Progesterone used in contraception and hormone therapy. Also known as Plan B, it's used as a single agent for emergency contraception, and as a contraceptive hormone released from the intrauterine device, known as the IUD.

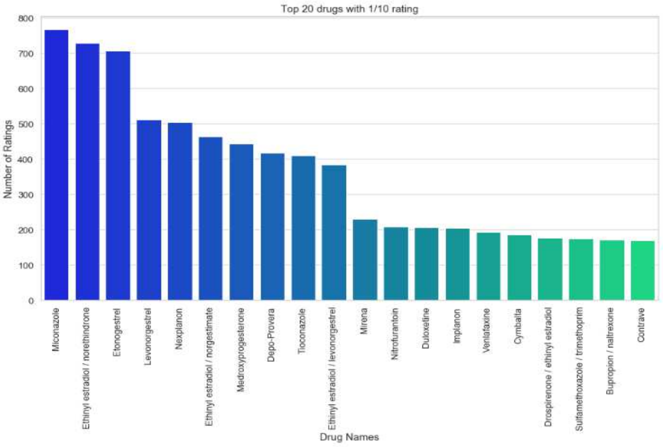

Fig 2 Top 20 drugs with 1/10 rating

Figure 2 is a bar graph that shows the top 20 drugs given in the data set with a rating of 1/10. 'Miconazole' is the drug with the highest number of 1/10 ratings, about 767. It's followed by 'Ethinyl estradiol/norethindrone' and 'Etonogestrel'.

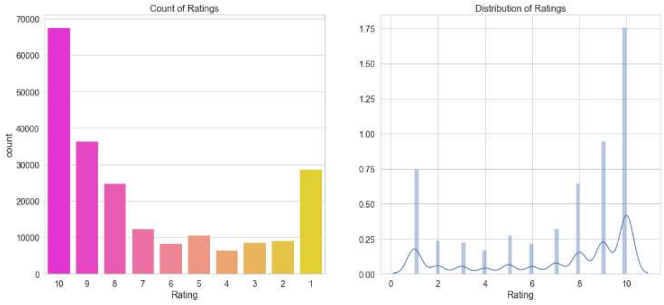

Fig 3 Count and Distribution of ratings

Figure 3 shows a distribution plot on the right-hand side and a bar graph of the same on the left-hand side. This shows the distribution of the ratings from 1 to 10 in the data set. It can be inferred that mostly it's 10/10 rating and after that 9 and 1. The data points with rating of the drugs from 2 to 7 is pretty low.

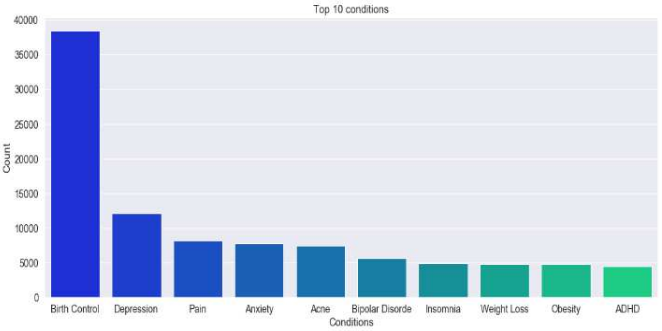

Fig 4 Top 10 Conditions in the Dataset

Figure 4 is a bar graph which exhibits the top 10 conditions the people are suffering from. In this data set 'Birth Control' is the most prominent condition by a very big margin followed by Depression and pain. The 'Birth Control' condition has occurred about 38,436 and the depressions have occurred about 12,164. It can easily be noticed that the 'Birth Control' is more than 3 time the depression in the whole data set. In the top 10 conditions, ADHD is in the 10th Position, ADHD stands for Attention deficit hyperactivity disorder.

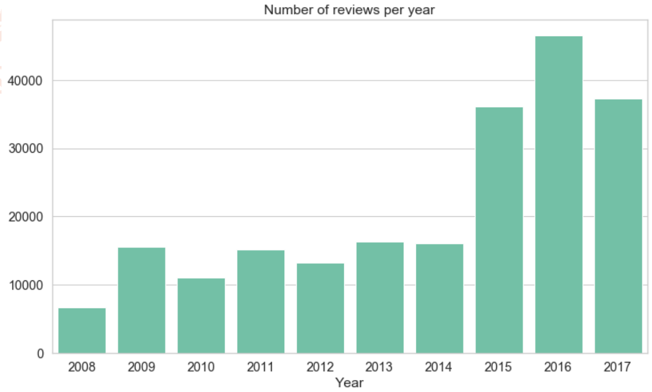

Fig 5 Number of reviews per year

Figure 5 is a Bar graph that shows the number of reviews in the data set per year. It can be inferred that most ratings are given in 2016 and 2008 has the least number of reviews. 2016 have 46507 reviews whereas 2008 have 6700 reviews.

A. Word Clouds

A Word Cloud is a visual representation of the frequency of the words occurring in the text or speech i.e., text data. It's also known as a tag cloud. Higher the frequency of the word in the text bigger its size will be and vice versa.

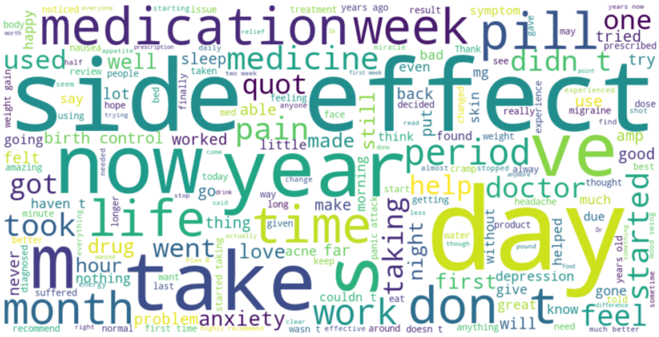

Fig 6 Word Cloud for the reviews with a rating of 10

Figure 6 shows the word cloud for the Drug reviews which has a rating of 10/10. We can see that words like 'side effect', 'now' and 'year' are occurring very frequently.

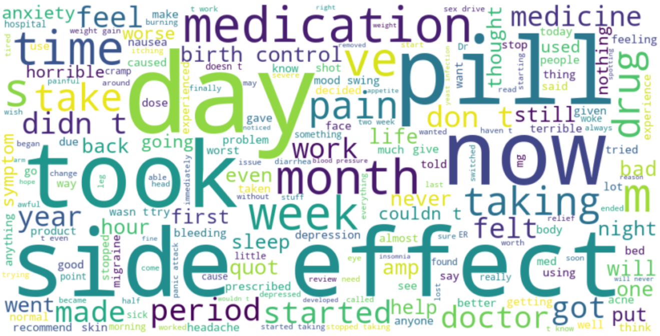

Fig 7 Word Cloud for the reviews with rating 1

Figure 7 shows the word cloud for the Drug reviews which has a rating of 1/10. We can see that words like 'side effect', 'pill' and 'period' are occurring very frequently.

IV. Preprocessing

Dataset contains 6 Features about the drugs which are 'drugName', 'condition', 'rating', 'date', 'usefulcount' and the 'review' itself. The sentiments of the reviews are not given so we have to generate them based on the ratings and use it as a target value which is to be predicted.

The train and test sets are merged so that we have a large combined dataset as the sentiments were not given in either of the sets. The size of the dataset is 215,063 Rows and 7 columns. The data set is then sorted based on the unique ID of the drugs (data points). The Data points with the null values in any of the given features are dropped as the dropped rows were only 0.55% of the total data. The shape of the dataset after dropping is (213,869, 7). The dates given in the dataset are not in the Date Time format, so I have changed it to the datetime64 format for further processing. The sentiment for the reviews is given based on the rating. If the rating is greater than 5 then it's positive sentiment and if the rating is less than or equal to 5 then it's a negative sentiment.

The reviews are cleaned before the feature engineering. Regular expressions are used to clean the reviews. The reviews are changed into the lower case first so that there's uniformity. After analyzing the reviews, it's found there's a repeating pattern "&#039;" which is occurring in most of the reviews hence they are removed. All the Special characters are removed. Some special characters were still left hence all the non-ASCII characters are removed. Trailing and leading whitespaces are removed from the reviews. Multiple whitespaces are replaced with a single space for more clarity.

The stopwords are also removed from the reviews as it'll be not very useful in the modelling. Only English stopwords are removed. The words in the review are also stemmed using the snowball stemmer. For example, the word running will be replaced with run.

V. Feature Engineering

There are initially 7 features given in the dataset which are 'drugName', 'condition', 'rating', 'date', 'usefulcount' and the 'review'. The heatmap of the correlation matrix of the numerical features is plotted before the feature engineering which is given in figure 8.

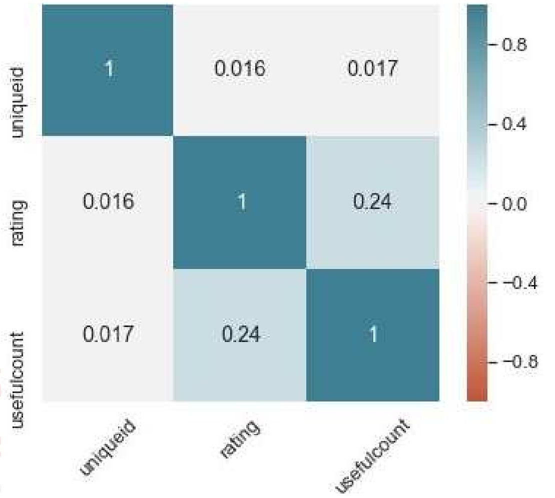

Fig 8 Correlation Heatmap before Feature Engineering

It's plotted with seaborn [6]. It can be inferred that the correlation between the 'usefulcount' and 'rating' is significant that is 0.24. 'uniqueID' is just the Unique ID given to each data point that is the consumer of the drug.

The textblob library is used to generate the sentiment polarity of the drug review [9]. This polarity is given to both the cleaned and uncleaned review. The interesting fact is that the correlation coefficient of the rating and the uncleaned review is 0.348 and with cleaned reviews is 0.233 hence it's greater for uncleaned review so, I have dropped the cleaned review columns and Cleaned it again but this time without removing the stopwords and stemming the words. Now the correlation coefficient of the cleaned review with the rating is 0.346 which is very good when compared to the last result.

The new features engineered are 'count_word' which is the number of words in each review, 'count_unique_word' which is the number of the unique words in the reviews. 'count_letters' is the letter count, 'punctuation_count' is the punctuation count, 'count_words_upper' is the upper-case word count,'count_words_title' is the title case word counts, 'count_stopwords' is the number of stop words in the review, and the 'mean_word_len' is the average length of the words in the review. The date is also divided into three columns which are day, month and year for separate features for training.

A new correlation heatmap is plotted using seaborn which contains all the new features engineered and the old features. It's given in figure 9.

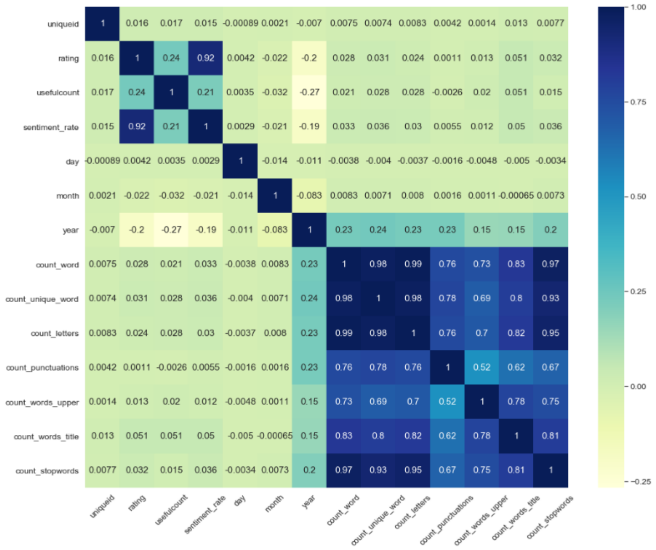

Fig 9 Correlation Heat map after Feature Engineering

The Label Encoder is used to change the categorical values of Drug Names and the conditions into numerical values for the machine learning modelling. There are 3,667 unique drugs in the dataset that's why One hot encoder is not used as it would generate 3,667 new features and it would be very computationally expensive.

VI. Modelling

The shape of the dataset after the deletion of the null values is (213,869, 7). 70% of the dataset is used for the training and the rest of the data i.e., 30% is used for the testing purpose. The shape of the training set is (149708, 15) and the shape of the test set is (64161, 15). It can be seen that before feature engineering that there were only 7 features but now there are 15. Three Machine learning models are trained which are LightGBM, XGBoost, and the CatBoost. The feature importance is also plotted for LightGBM and the description of these algorithms and their hyperparameters are given below. The feature importance given by the models XGBoost and LightGBM are also plotted.

A. LGBM

LightGBM stands for Light Gradient Boosting Machine. It's a gradient boosting framework which is based on the tree-based learning algorithms. It's a very efficient boosting algorithm. There are certain advantages for LGBM like fast training speed and high efficiency, lower memory usage and support of parallel and GPU learning as it is based on decision tree algorithms, it splits the tree leaf wise in accordance to the best fit. The LightGBM uses the XGBoost as a baseline. The LGBM algorithm outperforms many boosting algorithms in terms of efficiency and the size of the dataset it can comprehend easily. The hyperparameters of the LightGBM used are,

LGBMClassifier (n_estimators = 10000, learning_rate = 0.10, num_leaves = 30, subsample = .9, max_depth = 7, reg_alpha = .1, reg_lambda = .1, min_split_gain = .01, min_child_weight = 2, silent = -1, verbose = -1)

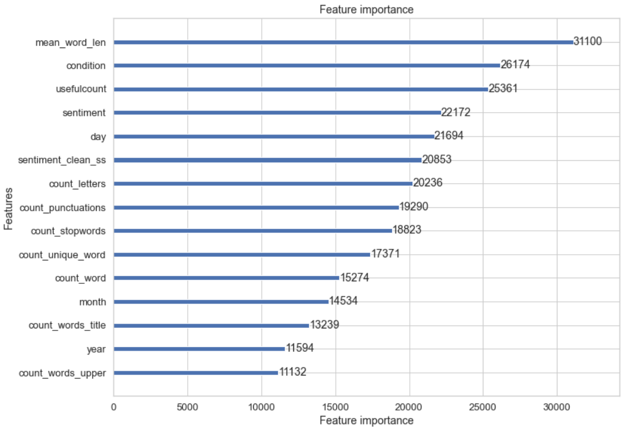

Fig 10 Feature Importance Plot by LGBM

Figure 10 depicts the feature importance plot using the LightGBM. It can be inferred that the most important feature in the dataset is the mean word length and after that the condition of the patient. The condition of the patient and the useful-count are very comparable in feature importance. The least important feature of them all is the upper-case word count.

B. XGBoost

XGBoost stands for extreme Gradient Boosting. XGBoost is a boosting algorithm used in many tasks in machine learning [2]. It is an optimized gradient boosting library which is basically designed to be highly efficient and flexible. It's also a Gradient Boosting framework which is under the machine learning algorithms. XGBoost bring forth the parallel tree boosting. It's open-sourced gradient boosting framework available for C++, Java, Python, R, Julia, Perl, and Scala. Most of the operating systems can be used for working on XGBoost. From the project description, it intends to produce a "Scalable, Portable and Distributed Gradient Boosting Library" [3].

Recently, XGBoost has earned a lot of popularity and became the choice of algorithm for many winning teams of machine learning competitions. It is an optimized Gradient Boosting Machine Learning library. The hyperparameters of the XGBoost are,

XGBClassifier (n_estimators = 10000, learning_rate = 0.10, num_leaves = 30)

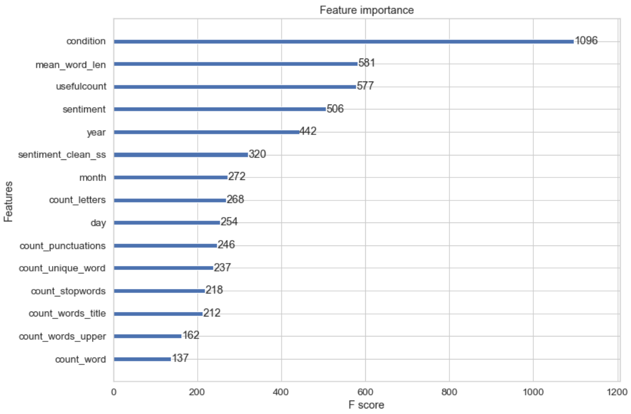

Fig 11. Feature Importance Plot by XGBoost

Figure 11 depicts the feature importance plot using the XGBoost. It can be inferred that the most important feature is the condition of the patient and it's far more important than the features following it. The features like sentiment, usefulcount and the year are equally important for the training.

C. CatBoost

CatBoost is an algorithm for gradient boosting on decision trees. It is developed by Yandex researchers and is used for many applications like search, recommendation systems, weather prediction and many other tasks at Yandex and in other companies well [4]. It's open-source as well. The hyperparameters of the CatBoost are,

CatBoostClassifier (iterations = 10000, learning_rate = 0.5)

VII. Evaluation Metrics

A. Accuracy

It is the ratio of the correct predictions i.e., the correct predicted values over the total prediction or total values.

Accuracy = (TP + TN) / (TP +TN + FP + FN)

B. Precision

Precision is defined as the ratio of true positive to the sum of true positive and false positive. It defines how often the classifier is correct when it predicts positive.

Precision = TP / (TP + FP)

C. Recall

Recall is defined as the ratio of true positive to the sum of true positive and false negative [1]. It defines how the classifier is correct for all positive instances.

Recall = TP / (TP + FP)

D. F1 Score

The F1 score can be interpreted or defined as a weighted average of the precision and recall as given in the equation, where an F1 score has its best value at 1 and worst score at 0.

F1 Score = 2(Precision*Recall) / (Precision + Recall)*

VIII. Results

Three machine learning models are trained which are LGBM, XGBoost and CatBoost. Given are some boosting algorithms in machine learning. The aim is to classify the sentiment of the drug reviews given by the patient as negative or positive. The results of the experiment are shown in table 1. It can be inferred that the best performing model is the LGBM followed by CatBoost. The accuracy of the LGBM is 88.89% with a good F1 Score of 0.922. The CatBoost algorithm also has a very good result and very close to LGBM. The XGBoost is not able to perform better in the task as compared to the other two models as the accuracy of the Model was 76.85%. Hence, LGBM is the best boosting algorithm in machine learning for the Drug review sentiment analysis.

Table I. Results

Algorithm	Accuracy	F1-Score	Precision	Recall
LGBM	0.888	0.922	0.902	0.942
XGBoost	0.768	0.846	0.786	0.917
CatBoost	0.882	0.916	0.904	0.929

IX. Conclusion

The main aim of the study is to predict the sentiment of the drug reviews given by the patients using the Boosting algorithms in Machine learning and compare them. Hence Exploratory Data Analysis was done to get more insight into the dataset and preprocessing was done to get the data ready for both the modelling and EDA. Initially, 7 features were given, hence feature engineering was done based on the EDA and reviews by the patients. The reviews were cleaned, and features are generated. The features were generated by both the cleaned and uncleaned reviews. In the Machine Learning modelling, three classification models were trained which were LightGBM, XGBoost, and the CatBoost. The performance metrics used here are Accuracy, F1-Score, Precision and Recall. The best performing model is the LGBM Classifier, but its accuracy and the classification report are comparable to the CatBoost Classifier. The accuracies were 0.888 and 0.882 respectively. The features importance is also plotted for LGBM and CatBoost. The XGBoost was not able to perform better than the other two.

References

[1] Duman, Ekrem & Ekinci, Yeliz & Tanriverdi, Aydin. (2012). Comparing alternative classifiers for database marketing: The case of imbalanced datasets. Expert Syst. Appl. 39. 48-53. 10.1016/j.eswa.2011.06.048.

[2] "XGBoost Hyperparameters Overview" [Online]. Available:https://www.numpyninja.com/post/xgboost-hyperparameters-overview.

[3] "XG Boost Documentation" [Online]. Available: https://xgboost.readthedocs.io/en/latest/index.html.

[4] "CatBoost Documentation" [Online]. Available: https://catboost.ai.

[5] "Matplotlib Documentation" [Online]. Available: https://matplotlib.org/stable/index.html.

[6] "Seaborn Documentation" [Online]. Available: https://seaborn.pydata.org/.

[7] Felix Gräßer, Surya Kallumadi, Hagen Malberg, and Sebastian Zaunseder. 2018. Aspect-Based Sentiment Analysis of Drug Reviews Applying Cross-Domain and Cross-Data Learning. In Proceedings of the 2018 International Conference on Digital Health (DH '18). ACM, New York, NY, USA, 121-125.

[8] Agrawal, A., Yang, X., Agrawal, R., Shen, X., & Menzies, T. (2020). Simpler Hyperparameter Optimization for Software Analytics: Why, How, When? *ArXiv, abs/2008.07334*.

[9] Madhu, Shrija. (2018). An approach to analyze suicidal tendency in blogs and tweets using Sentiment Analysis. International Journal of Scientific Research & Management Studies. 6. 34-36. 10.26438/ijsrcse/v6i4.3436.

[10] Na, Jin-Cheon & Kyaing, Wai. (2015). Sentiment Analysis of User-Generated Content on Drug Review Websites. Journal of Information Science Theory and Practice. 3. 6-23. 10.1633/JISTaP.2015.3.1.1.

[11] Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine-learn-ing techniques. Proceedings of the 2002 Conference on Empirical Methods in Natural Language Process-ing (pp. 79-86).

[12] Jo, Y., & Oh, A. H. (2011). Aspect and sentiment unification model for online review analysis. Proceed-ings of the Fourth International Conference on Web Search and Data Mining (WSDM) (pp. 815-824). Hong Kong.

[13] Ding, X., Liu, B., & Yu, P. S. (2008). A holistic lexi-22JISTaP Vol.3 No.1, 06-23con-based approach to opinion mining. Proceed-ings of the International Conference on Web Search and Web Data Mining (pp. 231-240). New York: ACM.

[14] Liu, B. (2012). Sentiment analysis and opinion mining. San Rafael, CA: Morgan & Claypool.