Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.777539

ANALYSING ACCURACY AND PRECISION OF SENTIMENT ANALYSIS OVER ENSEMBLE SUPERVISED LEARNING ALGORITHMS

Abhinav Gupta, Somil Kuchhal and Dr. Jayashree J*
School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, India

Abstract: Sentiment Analysis has become the go-to machine learning-based technique for understanding, decoding, and analysing millions of textual data items written by consumers. It is an efficient method to process subjective reviews into a computer-understandable language such as 1s and 0s. In today's research world, sentiment analysis has been tested over several different machine learning models but with our study, we have put a bunch of those models alongside each other and further used a Voting Classifier to determine the best. To increase the diversity and accuracy of our models, we have created an umbrella dataset constituting three popular datasets. We further created a more focused set of 75,000 entries from this larger dataset by developing a randomization algorithm. The supervised learning algorithms we have designed and tested are Random Forest, Extra Tree Classifier, Decision Tree, Logistic Regression, and XG Boosting. In our experiment, we were able to draw some good observations and conclusions. We used a word cloud library, often used for visual representation to further enhance the understanding of our data. Finally, we were able to conclude that the Extra Tree Classifier has the best accuracy compared to the other models while the Voting Classifier reported the best precision given the use case, we set up for this research study.

Keywords: Extra Tree Classifier, Machine Learning, Sentiment Analysis, Supervised Algorithms, Voting Classifier

1. Introduction

Our aim with this project research paper is to understand and quantifiably analyse the accuracy and precision achieved of sentiment analysis when performed over ensemble supervised machine learning algorithms. Sentiment analysis is a processing technique used to extract contextual information from user-based text reviews and comments.

As we know, in today's world, consumers know what they like and what they do not. Therefore, it has become overly critical for large scale industries to consider the consumer's feedback, reviews, and opinions on any product or service that they provide. However, at the same time, with the exponential rate at which the number of these text-based reviews is growing, it has become a huge challenge to be able to make sense of these reviews constructively without employing enormous amount of manpower. This is where sentiment analysis comes to shine. With this powerful solution, now large-scale sets of data with even hundreds of thousands of such consumer reviews or opinions can be easily processed into an understandable format with extremely limited time and resources required.

What makes our work unique is our approach to this problem. We scoured the internet and dug out three distinct datasets from different disciplines namely – the Twitter dataset containing tons of tweets and correspondingly a sentiment to describe them [23], a dataset of IMDB movie reviews which are very cluttered [24], and finally a dataset of amazon product reviews [25].

By combining these datasets, we formed a kind of superset and then used a self-designed randomization algorithm to select 75,000 rows to be used for training and testing our machine-learning models.

Another element of our research is the use of a Voting Classifier that takes into account all the models and uses the method of majority voting to produce the best overall output. The models being fed into the voting classifier are Random Forest, Extra Tree Classifier, Decision Tree, Logistic Regression, and XG Boosting. Each model is also separately designed to report the values of precision, recall as well as accuracy.

The paper has been organized in the following manner: Starting with Section II, we discuss the insights and learning we gained from referencing similar research work done in the field over the past few years. In Section III, we discuss our proposed approach to the problem along with figures, methods, algorithms, and implementation. In Section IV, we present the results obtained in our experiment, and finally, in Section V, we state our conclusions.

1. Related Work

A word convolutional approach was proposed by Z. Jianqiang et al. [1] based on unsupervised learning that utilizes latent contextual based semantical links as well as co-occurring statistical features among the tweet texts.

Whereas G. Li et al. [2] showcased a network model of sentiment information (SINM). The model's components were employed as the Transformer encoder and the LSTM. They were able to automatically find sentiment knowledge in Chinese text using a Chinese emotional lexicon.

However, neither of them used vectors in their models like X. Fan et al. [3]. That paper examined Sentiment Analysis with the efficiency of word vector representation in the challenge. Sentiment word extraction, sentiment word polarity identification, and text sentiment prediction were the three subtasks that they concentrated on most.

To understand the use of sentiment analysis in more focused use cases, we studied the work of V. Ikoro et al. [4] who portrayed the findings of a sentiment study conducted on Twitter by UK energy customers. By merging two different sentiment lexica-based functions, they further improved the accuracy of the findings.

Similarly, S. Vanaja and M. Belwal [5] made use of review data from Amazon to extract the concentrated phrases. They further identified each review's positivity, negativity, and neutrality through classification models.

M. Wongkar and A. Angdresey [6] used the Python programming language to develop an application for Twitter that perform sentiment analysis which of the tweets based on the 2019 Republic of Indonesia presidential candidates, which goes to show the real-world utility that this tool provides.

Another example of real-world use is the paper written by K. Zvarevashe and O. O. Olugbara [7] which claimed that the provided system of sentiment polarity automatically generates different sets of data for testing as well as training. These datasets are used to determine objective evaluations of the hotel's service based on the reviews provided.

Coinciding with one of our methodologies, X. Zhang, and X. Zheng [8] utilized TF-IDF to determine the weight of words using verbs, adjectives, and adverbs as text characteristics. Then they applied SVM and ELM with kernels to examine the text's emotional tendencies.

Y. Xu and Y. Ren [9] proposed a solution to the problem of differentiation among the commodity evaluation and commodity scoring by designing a self-updating iterative algorithm capable of performing sentiment analysis upon commodity evaluation text itself.

To gain perspective on the use of data pre-processing, we reviewed the work of Z. Jianqiang and G. Xiaolin [10] who explored the effects on sentiment analysis upon used of a text pre-processing approach in two different classification problems. They further used four classifiers and two feature algorithms to gather and process the data from five different Twitter based datasets.

K. Topal and G. Ozsoyoglu [11] claim to have a more efficient alternative to assess the scores and reviews given by movie reviewers. Assessment can be based on emotional content that creates an emotional mapping of the film by pooling and projecting it on the movie.

Similarly, tweets and product reviews also carry emotional content. M. Wöllmer et al. [12] focused on assessing the sentiment of the speakers automatically based on their video reviews of movies. A little offset from our domain, however, still quite helpful in explaining sentiment identification and reception. This technique involves incorporating audio characteristics, as utilized in emotion identification of speech data, and encoding of additional valence visual feature information expressed in the video, in addition to textual information.

The primary goal of the research of D. Mumtaz and B. Ahuja [13] was to introduce the Senti-lexical method to determine if a review is good, negative, or neutral. We also offered a strategy for dealing with terms that have a negative influence on the evaluations, and the significance of emoticons is examined.

Maier O. et al. [14] presented a novel method for automatically and reproducibly segmenting subacute ischemic stroke lesions in MR images in the presence of other diseases is provided. This new method was proposed based on the architecture built using the Extra Tree Forest machine learning voxel-wise classification model. It heavily relied on intensity-derived picture characteristics.

R. Shukla et al. [15] successfully showcased the use of ensemble regression methods such as Elastic Net, Ridge, Lasso, Multiple, Linear, etc. to predict total revenue being generated by businesses based on customer needs and business day traffic.

2. Proposed Algorithms

2.1. Random Forest

Random forest is commonly used in machine learning algorithms that join the output of more than one decision tree to a single decision tree. The model is often used in both Grouping and Relapse problems. When various classifiers are joined to work on the functioning of the model as well as the issue at hand, it is known as troupe learning and Random Forest depends on this very idea. It is easy to use, and it handles the classification and regression problem. The Random Forrest classifier is a web of several decision trees present of multiple subsets formed on the data that make use of the normal to determine the accuracy of that data. By considering predictions of each decision tree and refraining from dependence on a single choice tree, this model can predict the last result [17].

2.2. Extra Tree Classifier

A machine learning approach following the outfit learning method that trains many unlinked decision trees and collect the result from the group. However, there are many differences between extra tree and random forest when it is used to make a classification or

regression model. One of the key differences lies in the development of the decision trees in the forest. The first preparation test is responsible for developing every decision tree part of the Additional Tree Forest. Extra tree uses the whole data to train the decision tree. Moreover, it is relatively cheaper in cost comparison [16].

2.3. Decision Tree

The Decision Tree algorithm uses a tree format selection model. In fact, it turns out that trees have a lot in common and dominantly influence a large portion of the machine learning algorithmic functions including both the regression and classification type problems. When navigating through selections, selection trees can be used to explicitly reference selections and directions externally. However, information retrieval methods commonly used to determine how to achieve a specific target are also common in this domain [18].

2.4.Logistic Regression

A type of statistical based model wherein the variable is considered to be dichotomous (binary). It is used for analysing the categorical dependence variable using a given set of independent fixed values. This independent variable can be of the type nominal, interval based or even the ordinal type. Logistic Regression is same as Linear Regression except that how they work or used. It is also commonly referred to as the sigmoid function [19].

2.5. XG Boosting

XGBoost is an execution of Slope Supported decision trees. Boosting algorithms are ruling the machine learning community in terms of popularity and are also the most used models in Kaggle competitions, one of the largest machine learning communities online. In XG Boosting, a consecutive structure of decision trees is made. Loads are an integral part of this model which are used to predict results in the decision trees upon being appointed to each autonomous factor. The factors whose heaviness is miss-judged by the tree is then further expanded and catered by the next decision tree. It can do both classification and regression problems and achieve best result with minimum effort [20].

2.6. Voting Classifier

A vote classifier trains on hardware from different models and predicts the outcome (class) with the highest probability of choosing a class as a result. Basically, sums the findings of each classifier passed to the vote classifier and predicts the class of the result given the largest fraction of votes. The idea is that instead of creating separate specialized models and finding the accuracy of each, train these models and create separate models that predict revenue given the combined majority vote for each outcome class [21].

3. Implementation

In our research, we attributed 1 as the positive sentiment and 0 as the negative sentiment because we are aware that processing integers like 1s and 0s is much easier for the algorithm and the system as compared to processing strings, especially when it comes to such large datasets.

Moving on to working with our dataset. As briefly explained earlier, we have combined a total of three different datasets to form a big set from which we selected a group of 75,000 random distinct values. However, merging all these datasets present in different formats added a ton of pre-processing requirements that we needed to work on to use the data effectively.

Firstly, the data is converted to lower characters and all common contractions are expanded using a vast online dictionary. Then we remove the unnecessary URLs, stop words, special

characters, extra spaces, HTML tags, numbers, usernames, etc. from the data to make it much cleaner for the algorithms to read. Secondly, we use the method of Term Frequency-Inverse Document Frequency (TF-IDF) Vectorization. This process is popularly used to transform words into vectors which further helps us assign weight to them based on the frequency of their occurrence.

Another creative step we added to our process was the use of the PIL Library in python. This library takes our dataset as input and produces word clouds using the bag of words representation. These word clouds are visual representation that helps us gain a better understanding of what we are working with. Fig. 1 displays the positive word cloud and fig. 2 displays the negative word cloud for reference.

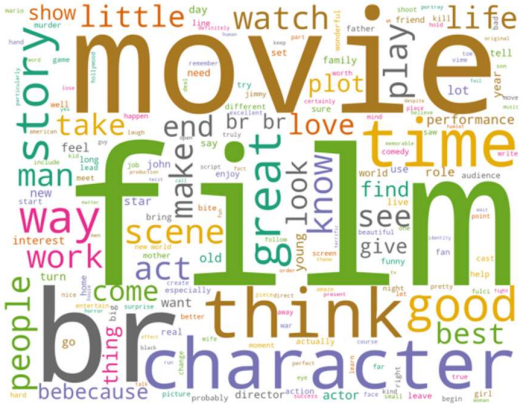

Fig.1. Positive Word Cloud

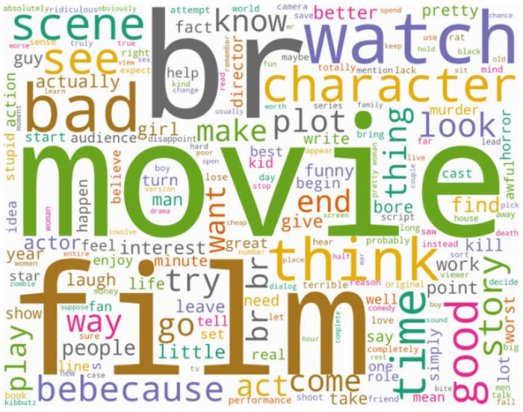

Fig.2. Negative Word Cloud

After completing the pre-processing and literature survey, we divided our dataset of 75,000 records into two segments, one for training and the other for testing since we are using supervised learning algorithms in this research study. Table 1 shows the division of records.

Table 1. Dataset Division for Implementation

Dataset Title	Number of Rows	Percentage
Training	56,250	75%

Testing	18,750	25%
Total	*75,000*	*100%*

Upon completing all the pre-experimental tasks, we moved on to the actual experiment. With the help of the sci-kit-learn library of python [22], we were able to design our own supervised machine learning algorithms for the models – Random Forest, Extra Tree Classifier, Decision Tree, Logistic Regression, and XG Boosting – that take our filtered and processed dataset as input and provides the precision, accuracy, and recall as output.

Once we were able to get the results from all the algorithms individually, we started building our voting classifier. First, we defined and evaluated our estimators which were all the models being pushed into the voting classifier engine. Next, we used a system of base majority voting to select the algorithm generating the best output. Finally, we print all our achieved results for review.

4. Results and Discussion

The most interesting part of the research comes now when we have the fruits of our labour in front of us and we can review them. Table 2 displays the list of all the algorithms we designed and tested. Corresponding to the models, we can see what precision, accuracy, and recall they were able to achieve.

Table 2. Results

Model	Accuracy	Precision	Recall
Decision Tree	77.506	75.858	78.943
XG Boosting	82.305	80.629	83.807
Random Forest	85.659	84.740	86.055
Logistic Regression	81.697	80.578	82.247
Extra Tree Classifier	86.545	85.249	87.522
Voting Classifier	86.504	85.527	87.019

Table 2 gives us a good understanding of our experiment's results. We can see that Extra Tree Classifier has the best accuracy among all the supervised machine learning models at approximately 86.545 and the best recall as well at approximately 87.522. The Extra Tree Classifier model has also achieved a precision score of 85.248 and stands at a close second just behind the Voting Classifier which has also reported a stellar accuracy and recall score. These two algorithms have produced the best scores out of all the ones we tested. However, it is important to note that Random Forest also performed extra-ordinarily with an accuracy of 85.659 (approx.), a mere 0.9 points behind our best accuracy score.

Just like the word cloud representation of our bag of words, we find graphical representation of data to be much easier to analyse and comprehend. Therefore, using the

matplotlib library of python, we plotted a few graphs based on the results obtained during our experiments.

Fig. 3 portrays a line graph where y axis denotes accuracy, and the x axis denotes all the supervised learning models we trained and tested. We can observe how the accuracies reported by the models fluctuate and finally reach its peak value at Extra Tree Classifier with a minute decline to the Voting Classifier. Fig. 4 represents a graph remarkably like fig. 3. It illustrates the precision produced by the algorithms and we can observe how the graph reaches its peak at the Voting Classifier after a slight increment from the Extra Tree model. Finally, in fig. 5 we can note the dissimilarity from the previous two graphs, primarily in the left half of the graph as we know XG Boosting reported a high recall score.

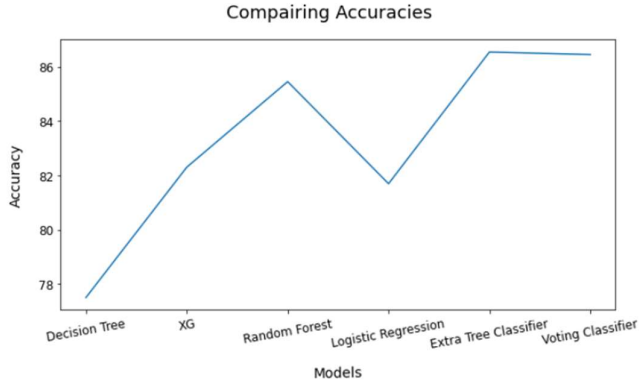

Fig. 3. Comparing Accuracy of Algorithms

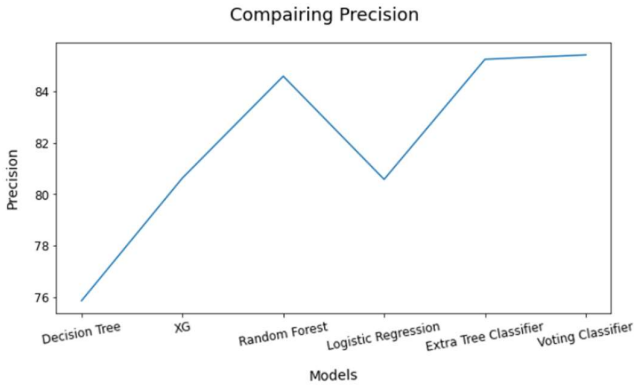

Fig. 4. Comparing Precision of Algorithms

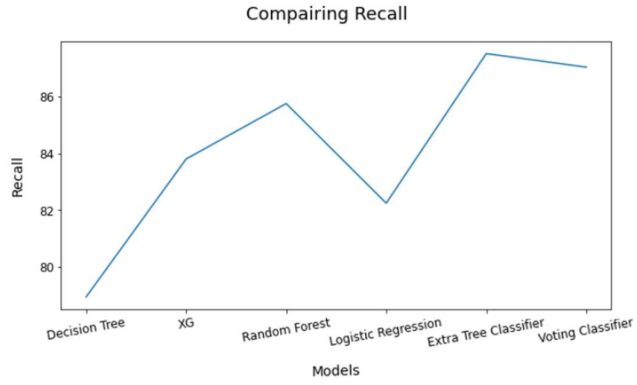

Fig. 5. Comparing Recall of Algorithms

5. Conclusion

In conclusion, our research study was able to demonstrate how comparing and analysing different supervised machine learning algorithms can help in choosing the best method for any use case. In our use case, where we merged reviews and opinions from different domains and then used the power of computer processing to build a filtration system to clean out that data, it is evident from the results and discussion that the Extra Tree Classifier performed the best. It is our understanding that this particular algorithm performed so well because of the way it works. Creating multiple decision trees and attacking each one randomly makes the algorithm predictions faster when compared to other ensemble-supervised machine learning algorithms due to their nature.

Secondly, the results show us that the Voting Classifier also reported amazing scores with the highest precision value and the second-highest accuracy and recall. The system of highest probability voting of all the model's predictions has allowed this algorithm to make predictions with pretty high accuracy. Other algorithms like Random Forest, XG Boosting, and Logistic Regression also reported scores that were consistent with our top performers. The accuracy ranged between 81.697 and 86.545, which is a gap of about 4.9 points. In the same way, the range of precision between these five algorithms is 80.578 to 85.527, maintaining a gap of about 5 points again. Decision Tree was the one algorithm in our set that performed a little below the mark with an accuracy of 77.506 approximately. The reason that this particular algorithm lagged is because of the use case we had set up. As explained previously, our dataset is diverse, and one of the key limitations of the Decision Tree model is its instability towards change. Even a small difference in the data can cause severe uncertainty in the processing, hence resulting in slightly lower performance.

It is important to note that the existing research study was used to create inputs and enhance system performance overall by identifying potential advantages and disadvantages. When surveys are compared, it becomes clear that Naive Bayes classifiers are used the majority of the time in research studies. Though it is a typical classification technique, it is a lot simpler approach with lesser accuracy and precision rates when compared to other related, relevant procedures.

References

[1] Kaggle. IMDB Dataset of 50K Movie Reviews. https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews

[2] Kaggle. Twitter Sentiment Analysis Dataset. https://www.kaggle.com/datasets/jp797498e/twitter-entity-sentiment-analysis?resource=download

[3] Kaggle. Amazon Reviews for Sentiment Analysis Dataset. https://www.kaggle.com/datasets/bittlingmayer/amazonreviews

[4] Z. Jianqiang, G. Xiaolin and Z. Xuejun, "Deep Convolution Neural Networks for Twitter Sentiment Analysis," in IEEE Access, vol. 6, pp. 23253-23260, 2018, doi: 10.1109/ACCESS.2017.2776930.

[5] G. Li, Q. Zheng, L. Zhang, S. Guo, and L. Niu, "Sentiment Information based Model for Chinese text Sentiment Analysis," 2020 IEEE 3rd International Conference on Automation,

Electronics and Electrical Engineering (AUTEEE), Shenyang, China, 2020, pp. 366-371, doi: 10.1109/AUTEEE50969.2020.9315668.

[6] X. Fan, X. Li, F. Du, X. Li, and M. Wei, "Apply word vectors for sentiment analysis of APP reviews," 2016 3rd International Conference on Systems and Informatics (ICSAI), Shanghai, China, 2016, pp. 1062-1066, doi: 10.1109/ICSAI.2016.7811108.

[7] V. Ikoro, M. Sharmina, K. Malik and R. Batista-Navarro, "Analyzing Sentiments Expressed on Twitter by UK Energy Company Consumers," 2018 Fifth International Conference on Social Networks Analysis, Management and Security (SNAMS), Valencia, Spain, 2018, pp. 95-98, doi: 10.1109/SNAMS.2018.8554619.

[8] S. Vanaja and M. Belwal, "Aspect-Level Sentiment Analysis on E-Commerce Data," 2018 International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 2018, pp. 1275-1279, doi: 10.1109/ICIRCA.2018.8597286.

[9] M. Wongkar and A. Angdresey, "Sentiment Analysis Using Naive Bayes Algorithm of The Data Crawler: Twitter," 2019 Fourth International Conference on Informatics and Computing (ICIC), Semarang, Indonesia, 2019, pp. 1-5, doi: 10.1109/ICIC47613.2019.8985884.

[10] K. Zvarevashe and O. O. Olugbara, "A framework for sentiment analysis with opinion mining of hotel reviews," 2018 Conference on Information Communications Technology and Society (ICTAS), Durban, South Africa, 2018, pp. 1-4.

[11] X. Zhang and X. Zheng, "Comparison of Text Sentiment Analysis Based on Machine Learning," 2016 15th International Symposium on Parallel and Distributed Computing (ISPDC), Fuzhou, China, 2016, pp. 230-233, doi: 10.1109/ISPDC.2016.39.

[12] Y. Xu and Y. Ren, "Research on Sentiment Analysis Model of Online Shopping Product Evaluation Based on Machine Learning," 2021 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), Dalian, China, 2021, pp. 482-486, doi: 10.1109/ICAICA52286.2021.9498066.

[13] Z. Jianqiang and G. Xiaolin, "Comparison Research on Text Pre-processing Methods on Twitter Sentiment Analysis," in IEEE Access, vol. 5, pp. 2870-2879, 2017, doi: 10.1109/ACCESS.2017.2672677.

[14] K. Topal and G. Ozsoyoglu, "Movie review analysis: Emotion analysis of IMDb movie reviews," 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), San Francisco, CA, USA, 2016, pp. 1170-1176.

[15] M. Wöllmer et al., "YouTube Movie Reviews: Sentiment Analysis in an Audio-Visual Context," in IEEE Intelligent Systems, vol. 28, no. 3, pp. 46-53, May-June 2013, doi: 10.1109/MIS.2013.34.

[16] D. Mumtaz and B. Ahuja, "Sentiment analysis of movie review data using Senti-lexicon algorithm," 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT), Bangalore, India, 2016, pp. 592-597, doi: 10.1109/ICATCCT.2016.7912069.

[17] Maier O., Wilms, M., von der Gablentz, J., Krämer, U. M., Münte, T. F., & Handels, H. (2015). Extra tree forests for sub-acute ischemic stroke lesion segmentation in MR sequences. Journal of neuroscience methods, 240, 89-100

[18] R. Shukla, P. Jindal, A. Gupta, and H. Y. Patil, "Total Revenue Prediction of a Sports Management Application: Grook Using Machine Learning Models," 2022 13th

International Conference on Computing Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2022, pp. 1-6, doi: 10.1109/ICCCNT54827.2022.9984472.

[19] Breiman, L. Random Forests. Machine Learning 45, 5–32 (2001).

[20] Maier O, Wilms M, von der Gablentz J, Krämer UM, Münte TF, Handels H. Extra tree forests for sub-acute ischemic stroke lesion segmentation in MR sequences. J Neurosci Methods. 2015 Jan 30; 240:89-100. doi: 10.1016/j.jneumeth.2014.11.011. Epub 2014 Nov 21. PMID: 25448384.

[21] Jijo, Bahzad & Mohsin Abdulazeez, Adnan. (2021). Classification Based on Decision Tree Algorithm for Machine Learning. Journal of Applied Science and Technology Trends. 2. 20-28.

[22] Karim F. Hirji, Cyrus R. Mehta & Nitin R. Patel (1987) Computing Distributions for Exact Logistic Regression, Journal of the American Statistical Association, 82:400, 1110-1117, DOI: 10.1080/01621459.1987.10478547

[23] Chen, Tianqi & Guestrin, Carlos. (2016). XGBoost: A Scalable Tree Boosting System. 785-794. 10.1145/2939672.2939785.

[24] Parhami, B. (1994). Voting algorithms. IEEE transactions on reliability, 43(4), 617-629.

[25] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.