Boosting and bagging classification for computer science journal

Nastiti Susetyo Fanany Putri [a,1], Aji Prasetya Wibawa [a,2,*], Harits Al Rasyid [a,3] , Andrew Nafalski [b,4], Ummi Rabaah Hasyim [c,5]

[a] Universitas Negeri Malang, Semarang st 5, Malang, 65145, Indonesia
[b] University of South Australia, Adelaide SA 5001, Australia
[c] Universiti Teknikal Malaysia Melaka,Durian Tunggal 76100, Malaka, Malaysia
[1] nastiti.susetyo.2005348@students.um.ac.id; [2] aji.prasetya.ft@um.ac.id ; [3] harits.al.ft@um.ac.id; [4] andrew.nafalski@unisa.edu; [5] ummi@utem.edu.my
* corresponding author

ARTICLE INFO

Article history
Selected paper from The 2022 5th International Symposium on Advanced Intelligent Informatics (SAIN'22), Yogyakarta (Virtually), September 14, 2022, http://sain.ijain.org/2022/. Peer-reviewed by SAIN'22 Scientific Committee and Editorial Team of IJAIN journal

Received August 31, 2022
Revised October 12, 2022
Accepted March 15, 2023
Available online March 31, 2023

Keywords
Journal classification
Ensemble learning
Boosting
Bagging
Decision tree
Gaussian naïve bayes
Scimago Journal Rank

ABSTRACT

In recent years, data processing has become an issue across all disciplines. Good data processing can provide decision-making recommendations. Data processing is covered in academic data processing publications, including those in computer science. This topic has grown over the past three years, demonstrating that data processing is expanding and diversifying, and there is a great deal of interest in this area of study. Within the journal, groupings (quartiles) indicate the journal's influence on other similar studies. SCImago provides this category. There are four quartiles, with the highest quartile being 1 and the lowest being 4. There are, however, numerous differences in class quartiles, with different quartile values for the same journal in different disciplines. Therefore, a method of categorization is provided to solve this issue. Classification is a machine-learning technique that groups data based on the supplied label class. Ensemble Boosting and Bagging with Decision Tree (DT) and Gaussian Nave Bayes (GNB) were utilized in this study. Several modifications were made to the ensemble algorithm's depth and estimator settings to examine the influence of adding values on the resultant precision. In the DT algorithm, both variables are altered, whereas, in the GNB algorithm, just the estimator's value is modified. Based on the average value of the accuracy results, it is known that the best algorithm for computer science datasets is GNB Bagging, with values of 68.96%, 70.99%, and 69.05%. Second-place XGBDT has 67.75% accuracy, 67.69% precision, and 67.83 recall. The DT Bagging method placed third with 67.31 percent recall, 68.13 percent precision, and 67.30 percent accuracy. The fourth sequence is the XGBoost GNB approach, which has an accuracy of 67.07%, a precision of 68.85%, and a recall of 67.18%. The Adaboost DT technique ranks in the fifth position with an accuracy of 63.65%, a precision of 64.21 %, and a recall of 63.63 %. Adaboost GNB is the least efficient algorithm for this dataset since it only achieves 43.19 % accuracy, 48.14 % precision, and 43.2% recall. The results are still quite far from the ideal. Hence the proposed method for journal quartile inequality issues is not advised.

1. Introduction

The advancement of computer science and digital technology are inextricably linked. Literacy resources must be readily available for scientists or students to innovate and perform research, which is how computer science has developed. Both print and digital media serve as the foundation for literacy.

Scientific publications are currently the most widely used literacy source. The journal contains this scientific article.

A journal is a collection of numerous articles in a specific scientific area [1]. Journal ranking websites can be used to assist in identifying high-quality publications. Journals are ranked according to the SCImago Journal Rank (SJR), where the higher the rank, the smaller the value of Q. Journals are ranked Q1, Q2, Q3, Q4, and Qn. Unfortunately, some journals mismatch the SJR indicators and the quartile label [2].

Data processing, such as categorization, is required to address this issue. Classification is a technique that seeks to anticipate models or functions that clarify and separate classes of ideas or facts. [3]. Similar studies have frequently employed a single classification approach across numerous disciplines. Therefore, the ensemble classification method will be used in this investigation. Comparing this method to the single method, the maximum outcomes tend to be higher [4].

Ensemble classification combines several classification algorithms with a voting system or other methods of grouping [5]. Conceptually trained individually to carry out the same task are the individual algorithms that make up the ensemble. The construction of ensembles can be either homogenous or heterogeneous [6]. Homogeneous ensembles are composed of a single basic method with at least two extra configurations or variations. For example, Bagging is a single basic method with multiple meta ensembles [7], [8], Boosting [7], [9], and stacking [10], [11]. The term "heterogeneous ensemble" refers to combining many model algorithms [12].

Any decision-making system's fundamental objective is to produce results with a high degree of accuracy. Therefore, this work aims to present the outcomes of the ensemble method's application to the classification of journal quartiles. We use this to enable the ranking system for the journal's articles to perform their job by providing proper predictions.

2. Method

2.1. Research Design

This study is divided into four stages. Data collection and processing are the first steps in research. The second phase involves preparing data to make it clean and appropriate for the categorization. The classification process comes next. Finally, a review was conducted. Fig. 1 depicts the order in which the various research steps were completed.

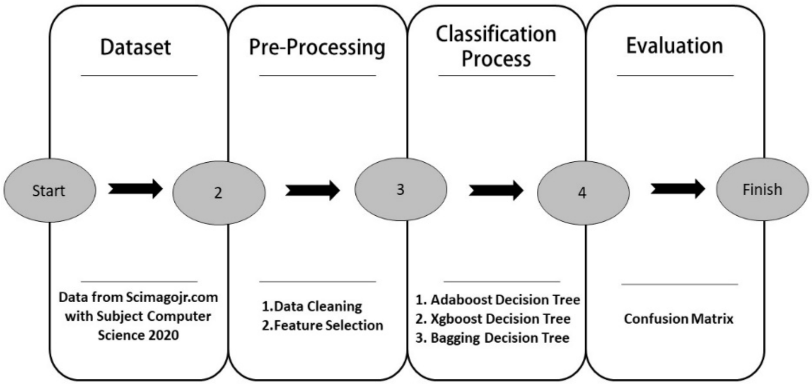

Fig. 1. Research design

2.2. Data Set

The SCImago Journal Rank is where secondary data for this study were gathered (SJR). Journal data related to computer science in 2020 is what was processed. The information was acquired on February 8, 2022, and is represented in Table. 1 as 1661 data points with 18 attributes. SJR Best Quartile is employed, and it denotes a multiclass classification with four classes: Q1, Q2, Q3, and Q4.

Table 1. Attributes dataset list

Attribute	Data Type	Ranges
Rank	Integer	1-1640
Sourceid	Real	1212500e+04-2110102e+10
Title	Nominal	Molecular system biology, journal of statistical software IEEE, etc
Type	Nominal	Journal
Issn	Nominal	1744492, 15487660, 2157846x, etc
SJR	Real	1.003-8.523
SJR Best Quartile	**Nominal**	**Q1, Q2, Q3, Q4, NQ**
H Index	**Integer**	**0-390**
Total Docs. (2020)	**Integer**	**0-18036**
Total Docs (3 Years)	**Integer**	**1-24267**
Total Refs.	**Integer**	**0-833161**
Total Cites. (3 years)	**Integer**	**0-116691**
Citable Docs. (3 years)	**Integer**	**1-24200**
Cite/ Doc (2 years)	**Real**	**0-69.560**
Ref.Doc.	Real	0-811
Country	Nominal	USA, Netherlands, UK, etc
Region	Nominal	Northern America, Western Europe, the Asiatic region, etc
Publisher	Nominal	Zhong gu, Springer New York, etc
Coverage	Nominal	1980-1986, 1971-2020, 2016-2020, etc
Categories	Nominal	Agriculture, Bioengineering, Computer Science, etc

Out of 18 qualities, only eight will be used in this study. Journal quartile class, H index, total documents (2020 and previous years), total references, total citations (3 years), citable documents (3 years), citations per document (2 years), and reference per document are the attributes that were employed. The SJR website uses this characteristic and shows it on its page; hence, it was picked. Therefore, these characteristics are expected to serve as the primary criteria for the journal quartiles classification. These characteristics are utilized to create journal quartile predictions, also called class labels, as independent variables [13].

2.3. Preprocessing

Preprocessing involves transforming raw data into a more understandable format [14]. Several preprocessing procedures are required to prepare the data before using the ensemble approach to classify. Data cleansing, integration, normalization, transformation, feature selection, and other preprocessing methods are frequently employed [15], [16]. The technique applied in this case is data cleaning and feature selection. Cleaning data sets by eliminating data that has no value (missing value) or noise is the goal of data cleaning [17]. The dataset used has 1661 data before cleaning, to 1640 instances after data cleansing. Table. 2 contains details of the dataset after cleansing.

Table 2. Dataset after cleaning

SJR Best Quartile	Sum
Q1	445
Q2	416
Q3	417
Q4	362
Sum	1640

Feature selection is only used to determine the variables used in the classification process. The attributes used as independent variables are H index and Total Docs. (2020), Total Docs. (3 years), Total Refs, Total Cites (3 years), Citable Docs. (3 years), Cites / Doc. (2 years), and Ref. / Docs. SJR best quartile is used as the dependent variable or target class.

2.4. Classification Process

The procedure of classification is the most crucial phase. The meta-ensemble for boosting and bagging is used in this research. Boosting, also known as hypothesis boosting, combines weak algorithms to create a stronger algorithm [18]. The probably approximately correct (PAC) learning model, a theoretical framework for the study of machine learning, is where boosting got its start [19]. Boosting is intended to fix inaccurate predictions [20]. To correct the first model's predictions, the models are fitted and introduced to the ensemble one at a time. The third model then attempts to correct the predictions of the second model, and so on. Fig. 2 is how boosting work is illustrated. Adaboost and XGboost are the only two boosting types used in this study. The two meta-ensembles were selected since they are most frequently applied in ensemble research [21]. Moreover, boosting is an easy-to-read and interpret algorithm.

Adaboost (Adaptive Boosting) begins by making predictions on the original dataset that are as straightforward as possible before assigning equal weights to each observation [22]. If the prediction produced using the first learner is erroneous, it gives the incorrectly predicted statement more importance and goes through an iterative loop. Meanwhile, Extreme Gradient Boosting, or XGBoost, is an ensemble technique that combines a decision tree and boosted gradient [23]. The model's effectiveness and computation speed are key factors in this approach [24]. Thus, parallelization, distributed computing, cache optimization, and out-of-core computing are possible. To determine the best split, XGBoost employs both a pre-sorted algorithm and a histogram-based technique [25]. All of the data points for a feature are divided into discrete bins using the histogram-based technique, which then uses these discrete bins to calculate the split value of the histogram. Additionally, with XGBoost, the trees' number of terminal nodes can vary, and the left weights of the trees whose calculations use less evidence are decreased more severely.

Multiple learners are combined by bagging (bootstrap aggregation) to lower predicted variance [26]. An ensemble that is set up in parallel is a bag. To put it simply, Bagging is a random forest method with more trees [27]. Fig. 3 illustrates how Bagging operates. The decision tree is the learning foundation for the Bagging applied in this work. The ensemble's classifiers are created using various random samples from the training set [28]. Using the extension, bagging ensembles help to lower variance and prevent overfitting [29]. Bagging involves the two phases of bootstrapping and aggregation [30]. Bootstrap resampling the training dataset to create a sub-dataset from datasets [31]. An aggregating step combines various predictions into one final value [32].

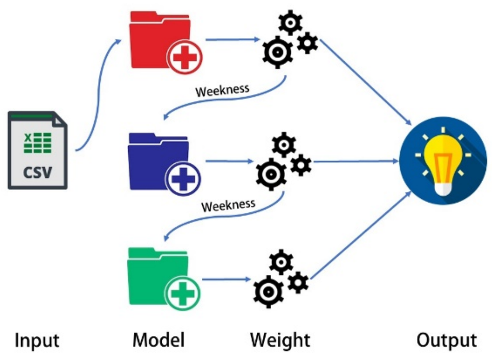

Fig. 2. Boosting ensemble

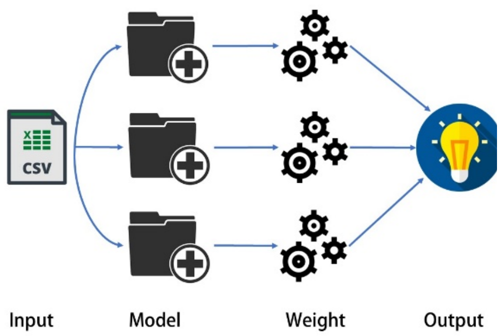

Fig. 3. Bagging ensemble

In this research, decision tree (DT) and Gaussian Naïve Bayes are applied as the base learner in both ensemble methods.

2.5. Evaluation

The Confusion Matrix (CM) method of determining accuracy outcomes is used in this ensemble classification investigation [33]. The One-Against-All technique was used to apply CM as an assessment model. This calculation takes accuracy, precision, and recall into account. CM is displayed in Table. 3.

Table 3. Confusion matrix

Class	Positive Classified	Negative Classified
Positive	TP (True Positive)	FN (False Positive)
Negative	FP (False Positive)	TN (True Negative)

3. Results and Discussion

This project uses meta-ensemble boosting and bagging to categorize data from computer science journals. Adaboost and XGBoost are two examples of meta ensembles used for boosting. Gaussian Naïve Bayes and Decision trees are the weak learning algorithms employed. The variables n depth and n estimator, which describe the number of branching in the final tree and the number of times the algorithm is repeated in a single ensemble voyage, are modified in the decision tree application. Only the number of times the fundamental method (n estimators) is repeated changes in the Naïve Bayes scenario.

3.1. Boosting Classification

The Adaboost meta-ensemble decision tree boosts the first classification method. N depth undergoes four changes, with the most considerable value at 5, 10, 15, and 50. Experiments were conducted with various estimation values, including 10, 50, 100, 150, 200, and 250 for each depth value. The objective was to ascertain how changes in depth and estimator affected the classification results for accuracy, precision, and recall. Adaboost and Bagging meta, which employ the decision tree algorithm as the basis learner use this method. The outcomes of the Boosting classification method are shown in Table. 4.

Table 4. Boosting classification result

Method	N_Depth	N_Estimators	Accuracy	Precision	Recall
Adaboost Decision Tree	5	10	48.318	52.428	48.665
		50	63.17	64.774	63.047
		100	63.62	65.833	63.146
		150	63.17	64.774	63.047
		200	65.548	67.182	65.282
		250	**65.79**	**67.103**	**65.486**
	10	10	64.755	64.9	64.86
		50	66.949	67.28	66.93
		100	**68.05**	**68.29**	**67.98**
		150	67.99	68.254	67.926
		200	67.774	68.007	67.691
		250	67.924	68.205	67.826
	15	10	63.901	63.601	63.917
		50	65.55	65.416	65.504
		100	65.123	65.058	65.109
		150	65.305	65.3	65.297
		200	**66.552**	**66.621**	**66.441**
		250	65.7	65.776	65.648
	50	10	60.12	60.044	60.272
		50	60.427	60.477	60.567
		100	60.425	60.381	60.593
		150	60.579	60.532	60.705
		200	**60.761**	**60.69**	**60.888**
		250	60.213	60.144	60.345

Table 4. (Cont.)

Method	N_Depth	N_Estimators	Accuracy	Precision	Recall
XGboost Decision Tree	5	10	69.51	69.36	69.68
		50	69.21	69.56	69.29
		100	**70.73**	**70.69**	**70.82**
		150	69.51	69.35	69.72
		200	70.12	70.07	70.23
		250	69.51	69.39	69.57
	10	10	66.77	66.73	66.87
		50	66.77	66.79	66.86
		100	66.77	66.92	66.85
		150	66.46	66.58	66.51
		200	67.07	67.09	67.14
		250	**67.38**	**67.46**	**67.35**
	15	10	66.16	65.95	66.46
		50	66.16	65.96	66.4
		100	**67.38**	**67.31**	**67.57**
		150	67.07	67.22	67.2
		200	66.46	66.56	66.55
		250	65.55	65.63	65.67
	50	10	66.46	66.17	66.68
		50	67.38	67.14	67.39
		100	68.59	68.45	68.53
		150	68.59	68.41	68.5
		200	**68.59**	**68.37**	**68.54**
		250	67.68	67.49	67.59

Table 4 shows that the accuracy results obtained with the XGBoost meta-ensemble typically outperform those obtained with Adaboost. However, because the decision tree utilized as the weak learner has a low level of performance stability, the accuracy value also varies. A linearity test was conducted to ascertain the impact of the depth and estimator values, and the results are shown in Fig. 4 and Fig. 5. As observed in the image, the test results show a linear relationship between the rise in estimator value and the depth of the decision tree algorithm.

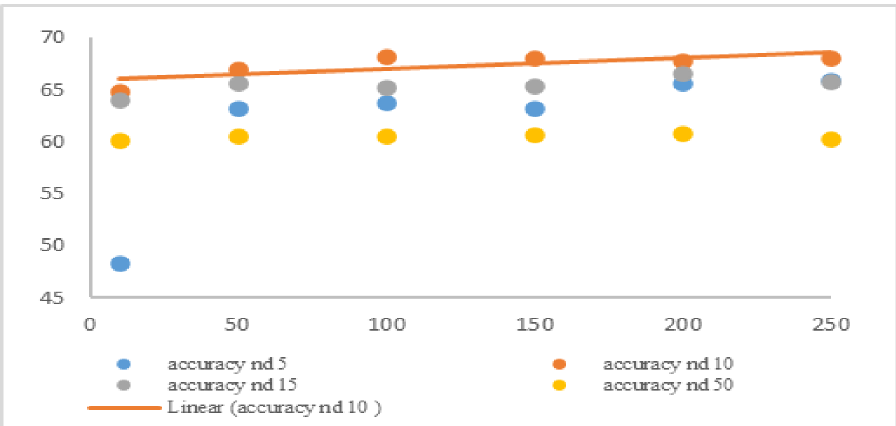

Fig. 4. Effect of estimation on classification performance

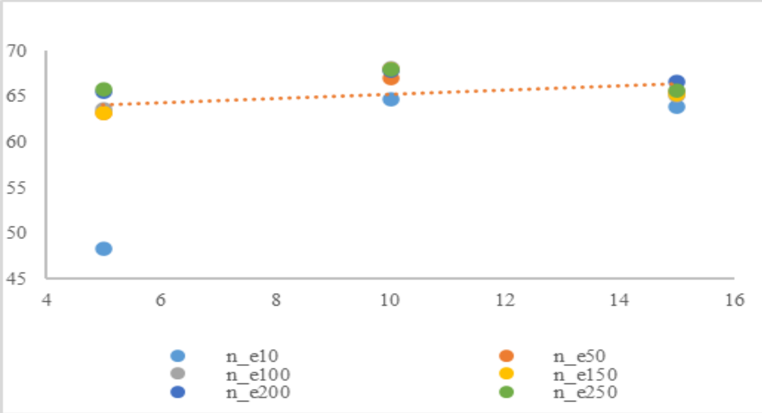

Fig. 5. Effect of depth on classification performance

Utilizing the Gaussian Naïve Bayes type, Naïve Bayes is the second subpar algorithm. Because Naïve Bayes is based on Bayes' theory, which holds that each predictor is unique, only modifications to the value of n estimators are changed in this stage as opposed to the decision tree [24]. The results of the improved Naïve Bayes performance in the journal quartiles categorization are displayed in Table 5. The employment of naïve Bayes meta-ensemble XGBoost yields significantly higher accuracy than the Adaboost approach, similar to the prior stage. On AdaBoost, the smallest estimator value generates the worst accuracy value, whereas, on XGBoost, the converse is true.

Table 5. Boosting gausian naïve bayes classification result

Method	N_estimator	Accuracy	precision	Recall
Adaboost Gaussian Naïve Bayes	10	36.59	45.77	37.31
	50	46.34	47.75	46.82
	100	43.29	46.66	43.55
	150	44.51	51	43.99
	200	**46.34**	**50.85**	**45.19**
	250	42.07	46.79	42.31
XGBoost Gausian Naïve Bayes	**10**	**68.59**	**68.56**	**68.7**
	50	67.99	67.91	67.98
	100	66.46	66.16	66.55
	150	66.77	66.48	66.89
	200	66.16	65.85	66.33
	250	66.46	66.16	66.61

A linearity test was also carried out to ascertain the impact of variations in the number of n estimators on the classification work's outcomes. The linearity test results for bossing Naïve Bayes are shown in Fig. 6.

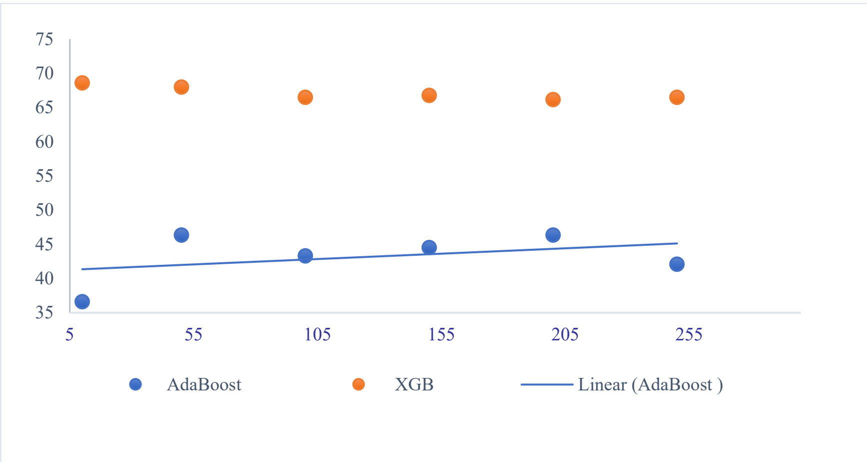

Fig. 6. Boosting Naïve Bayes Accuracy Linear Test

In Naïve Bayes, the change in the estimator value is linear, but it is not very significant.

3.2. Bagging

The ensemble bagging technique is used for the second categorization. Using Decision Tree and Gaussian Naïve Bayes as weak algorithms is similar to how the previous technique was applied. Table. 6 and Table. 7 for the decision tree and Gaussian Naïve Bayes, respectively, show the outcomes of both methods.

Table 6. Bagging Decision Tree Classification Result

Method	N_Depth	N_Estimators	Accuracy	precision	Recall
Bagging Decision Tree	5	10	69.21	71.55	69.34
		50	**69.51**	**71.42**	**69.87**
		100	69.82	71.37	70.11
		150	68.9	70.56	69.15
		200	69.21	70.92	69.49
		250	69.51	71.33	69.79
	10	10	65.24	66.66	65.12
		50	65.55	66.69	65.61
		100	66.46	66.84	66.47
		150	66.16	67.07	66.18
		200	67.38	68.04	67.29
		250	**67.68**	**68.21**	**67.61**
	15	10	65.25	65.89	64.97
		50	67.07	67.15	67.03
		100	67.07	67.35	67.05
		150	67.07	67.28	66.99
		200	**67.99**	**68.04**	**67.91**
		250	67.98	68.31	67.93
	50	10	64.33	65.29	64.11
		50	66.46	66.51	66.43
		100	66.46	66.76	66.35
		150	65.85	66.06	65.74
		200	67.38	67.59	67.29
		250	**67.68**	**68.21**	**67.61**

Table 7. Bagging Gausian Naïve Bayes Classification Result

Method	N_Estimators	Accuracy	Precision	Recall
Bagging Gaussian Naïve Bayes	10	67.68	68.21	67.61
	50	**69.21**	**71.55**	**69.34**
	100	69.21	71.55	69.34
	150	69.21	71.55	69.34
	200	69.21	71.55	69.34
	250	69.21	71.55	69.34

The two tables show that Naïve Bayes may achieve the same accuracy value with a relatively low estimation value, whereas the decision tree method tends to do so with a higher estimation value. The outcomes of the decision tree testing with ensemble bagging approaches are shown in Fig. 7 and Fig. 8.

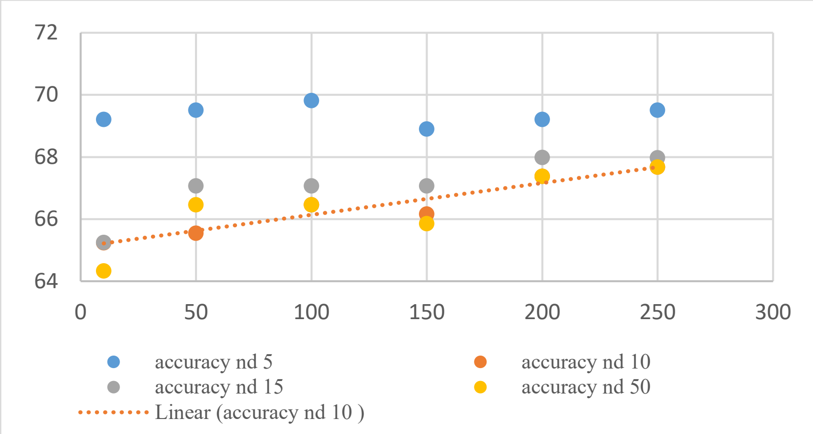

Fig. 7. Effect of Estimation on DT Bagging Classification Performance

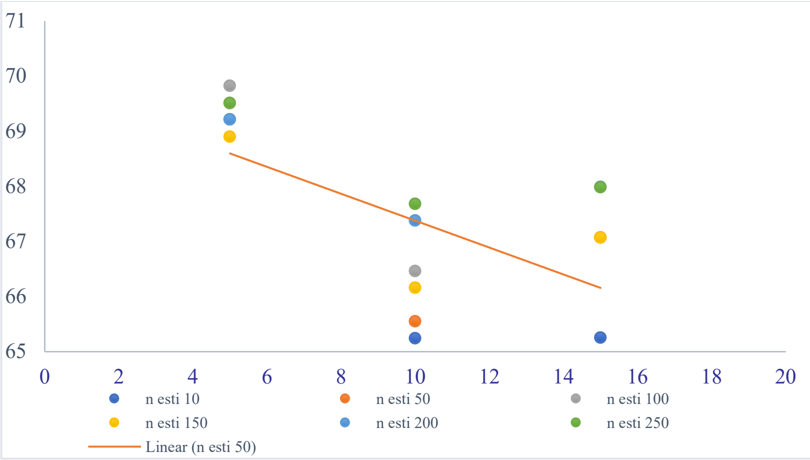

Fig. 8. Effect of Depth on DT Bagging Classification Performance

The predicted value and depth value also impact the bagging performance on the base learner decision tree. However, there is an inverse link to in-depth Bagging, where the accuracy decreases as the depth value increases. The increase in the anticipated value of Naïve Bayes bagging is illustrated in Fig. 9 bellow. When the graph reaches a particular point, it starts to slope.

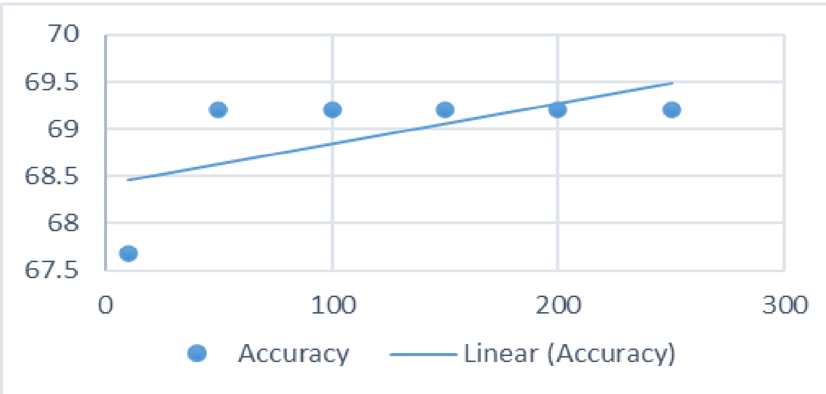

Fig. 9. Bagging Gausian Naïve Bayes Classification Performance

3.3. Comparation Ensemble and Single Classifier

DT and GNB are utilized as classifier algorithms. Table. 8 displays the results of using the method. These results serve as a contrast to the ensemble method that was utilized.

Table 8. Single classifier result

Algorithm	Accuracy	Precision	Recall
Decision Tree	57.52	57.57	57.69
Gaussian Naïve Bayes	48.17	51.09	49.74

The DT algorithm is the most accurate and stable regarding precision and recall. Table. 9 displays a comparison of the ensemble and single classifier values. The ensemble values listed are the top results from all completed experiments.

Table 9. Result Comparison

Algorithm	Single Classifier			Ensemble Classifier								
				Bagging			Boosting					
							Adaboost			*XGBoost*		
	Acc.	*Prec.*	*Rec.*	*Acc.*	*Prec.*	*Rec.*	*Acc.*	*Prec.*	*Rec.*	*Acc.*	*Prec.*	*Rec.*
Decision Tree	57.52	57.57	57.69	69.51	71.42	69.87	68.05	66.62	66.44	**70.73**	**70.69**	**70.82**
Gaussian Naïve Bayes	48.17	51.09	49.74	69.21	71.55	69.34	46.34	50.85	45.19	68.59	68.56	68.7

The GNB Bagging method, which saw a value gain of 21.04%, is reported to have experienced the largest value growth. Using the baseline method, XGBoost GNB comes in second with a difference of 20.42%. Third-placed XGBoost DT has grown by 13.21%. Bagging DT, with a differential of 11.99%, comes in fourth. AdaBoost is an ensemble method with the least increase (10.52%) and the lowest AdaBoost GNB accuracy (-1.83%). Overall, however, the ensemble method outperforms the single method in terms of accuracy.

4. Conclusion

The bagging approach is the ensemble method that performs better in the quartile categorization of computer science subject journals, as inferred from the results' description and discussion. As a result of their around 60 percent average accuracy, decision tree and Naïve Bayes work nearly equally well for ineffective methods. The decision tree algorithm's value of n depth affects classification performance; the larger the value of n depth, the greater the chance of obtaining the highest accuracy. The classification accuracy value is positively impacted by adding the predicted values for the decision tree and Naïve Bayes algorithms. Therefore, it is strongly advised that high estimation values be provided in comparable research in the future. However, this study's accuracy tends to vary, necessitating additional investigation to enhance it. Incomparable investigations in the future, it is strongly advised to use different meta-ensemble methods like stacking or other algorithms with a considerably better stability value.

Acknowledgment

This paper and the research behind it would not have been possible without the hard work of our team, as well as financial support from the State University of Malang (UM) through a research grant, PPM 2022.

Declarations

Author contribution. All authors contributed equally as the main contributor of this paper. All authors read and approved the final paper.
Funding statement. This research is funded by Universitas Negeri Malang, Hibah Publikasi Tesis 2022, Lembaga Penelitian dan Pengabdian Masyarakat.
Conflict of interest. The authors declare no conflict of interest.
Additional information. No additional information is available for this paper.

References

[1] J. Vom Brocke, R. Winter, A. Hevner, and A. Maedche, "Special issue editorial – accumulation and evolution of design knowledge in design science research: A journey through time and space," *J. Assoc. Inf. Syst.*, vol. 21, no. 3, pp. 520–544, 2020, doi: 10.17705/1jais.00611.

[2] A. P. Wibawa *et al.*, "Naïve Bayes Classifier for Journal Quartile Classification," *Int. J. Recent Contrib. from Eng. Sci. IT*, vol. 7, no. 2, p. 91, 2019, doi: 10.3991/ijes.v7i2.10659.

[3] M. Sabokrou, M. Fathy, G. Zhao, and E. Adeli, "Deep End-to-End One-Class Classifier," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 32, no. 2, pp. 675–684, 2021, doi: 10.1109/TNNLS.2020.2979049.

[4] M. Abdar *et al.*, "A new nested ensemble technique for automated diagnosis of breast cancer," *Pattern Recognit. Lett.*, vol. 132, pp. 123–131, 2020, doi: 10.1016/j.patrec.2018.11.004.

[5] J. Nalić, G. Martinović, and D. Žagar, "New hybrid data mining model for credit scoring based on feature selection algorithm and ensemble classifiers," *Adv. Eng. Informatics*, vol. 45, p. 101130, 2020, doi: 10.1016/j.aei.2020.101130.

[6] J. L. Fernández-Alemán, J. M. Carrillo-de-Gea, M. Hosni, A. Idri, and G. García-Mateos, "Homogeneous and heterogeneous ensemble classification methods in diabetes disease: a review," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2019, pp. 3956–3959, doi: 10.1109/EMBC.2019.8856341.

[7] Y. Himeur, A. Alsalemi, F. Bensaali, and A. Amira, "Robust event-based non-intrusive appliance recognition using multi-scale wavelet packet tree and ensemble bagging tree," *Appl. Energy*, vol. 267, p. 114877, 2020, doi: 10.1016/j.apenergy.2020.114877.

[8] G. Tuysuzoglu and D. Birant, "Enhanced bagging (eBagging): A novel approach for ensemble learning," *Int. Arab J. Inf. Technol.*, vol. 17, no. 4, pp. 515–528, 2020, doi: 10.34028/iajit/17/4/10.

[9] X. Huang *et al.*, "Ensemble-boosting effect of Ru-Cu alloy on catalytic activity towards hydrogen evolution in ammonia borane hydrolysis," *Appl. Catal. B Environ.*, vol. 287, p. 119960, 2021, doi: 10.1016/j.apcatb.2021.119960.

[10] A. Mosavi, F. Sajedi Hosseini, B. Choubin, M. Goodarzi, A. A. Dineva, and E. Rafiei Sardooi, "Ensemble Boosting and Bagging Based Machine Learning Models for Groundwater Potential Prediction," *Water Resour. Manag.*, vol. 35, no. 1, pp. 23–37, 2021, doi: 10.1007/s11269-020-02704-3.

[11] Y. Xiong, M. Ye, and C. Wu, "Cancer Classification with a Cost-Sensitive Naive Bayes Stacking Ensemble," *Comput. Math. Methods Med.*, vol. 2021, 2021, doi: 10.1155/2021/5556992.

[12] B. A. Hassan and T. A. Rashid, "A multidisciplinary ensemble algorithm for clustering heterogeneous datasets," *Neural Comput. Appl.*, vol. 33, no. 17, pp. 10987–11010, 2021, doi: 10.1007/s00521-020-05649-1.

[13] A. P. Wibawa, "International Journal Quartile Classification Using the K-Nearest Neighbor Method," 2019, doi: 10.1109/ICEEIE47180.2019.8981413.

[14] K. Nahar, B. I. Shova, T. Ria, H. B. Rashid, and A. H. M. S. Islam, "Mining educational data to predict students performance," *Educ. Inf. Technol.*, vol. 26, no. 5, pp. 6051–6067, Sep. 2021, doi: 10.1007/s10639-021-10575-3.

[15] H. Benhar, A. Idri, and J. L. Fernández-Alemán, "Data preprocessing for heart disease classification: A systematic literature review," *Comput. Methods Programs Biomed.*, vol. 195, p. 105635, 2020, doi: 10.1016/j.cmpb.2020.105635.

[16] S.-A. N. Alexandropoulos, S. B. Kotsiantis, and M. N. Vrahatis, "Data preprocessing in predictive data mining," *Knowl. Eng. Rev.*, vol. 34, p. e1, 2019, doi: 10.1017/S026988891800036X.

[17] K. H. Tae, Y. Roh, Y. H. Oh, H. Kim, and S. E. Whang, "Data Cleaning for Accurate, Fair, and Robust Models: A Big Data - AI Integration Approach," 2019, doi: 10.1145/3329486.3329493.

[18] C. B. C. Latha and S. C. Jeeva, “Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques,” *Informatics Med. Unlocked*, vol. 16, p. 100203, 2019, doi: 10.1016/j.imu.2019.100203.

[19] R. Sambasivan, S. Das, and S. K. Sahu, “A Bayesian perspective of statistical machine learning for big data,” *Comput. Stat.*, vol. 35, no. 3, pp. 893–930, Sep. 2020, doi: 10.1007/s00180-020-00970-8.

[20] H. Liu, R. Yang, T. Wang, and L. Zhang, “A hybrid neural network model for short-term wind speed forecasting based on decomposition, multi-learner ensemble, and adaptive multiple error corrections,” *Renew. Energy*, vol. 165, pp. 573–594, Mar. 2021, doi: 10.1016/j.renene.2020.11.002.

[21] H. Jafarzadeh, M. Mahdianpari, E. Gill, F. Mohammadimanesh, and S. Homayouni, “Bagging and Boosting Ensemble Classifiers for Classification of Multispectral, Hyperspectral and PolSAR Data: A Comparative Evaluation,” *Remote Sens.*, vol. 13, no. 21, 2021, doi: 10.3390/rs13214405.

[22] B. So, J.-P. Boucher, and E. A. Valdez, “Cost-sensitive Multi-class AdaBoost for Understanding Driving Behavior with Telematics,” Jul. 2020, doi: 10.48550/arXiv.2007.03100.

[23] E. K. Sahin, “Assessing the predictive capability of ensemble tree methods for landslide susceptibility mapping using XGBoost, gradient boosting machine, and random forest,” *SN Appl. Sci.*, vol. 2, no. 7, p. 1308, Jul. 2020, doi: 10.1007/s42452-020-3060-1.

[24] M. Ma *et al.*, “XGBoost-based method for flash flood risk assessment,” *J. Hydrol.*, vol. 598, p. 126382, Jul. 2021, doi: 10.1016/j.jhydrol.2021.126382.

[25] V. Patel, S. Choe, and T. Halabi, “Predicting Future Malware Attacks on Cloud Systems using Machine Learning,” in *2020 IEEE 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*, May 2020, pp. 151–156, doi: 10.1109/BigDataSecurity-HPSC-IDS49724.2020.00036.

[26] F. Harrou, A. Saidi, and Y. Sun, “Wind power prediction using bootstrap aggregating trees approach to enabling sustainable wind power integration in a smart grid,” *Energy Convers. Manag.*, vol. 201, p. 112077, 2019, doi: 10.1016/j.enconman.2019.112077.

[27] D. C. Yadav and S. Pal, “Prediction of thyroid disease using decision tree ensemble method,” *Human-Intelligent Syst. Integr.*, vol. 2, no. 1–4, pp. 89–95, 2020, doi: 10.1007/s42454-020-00006-y.

[28] M. Ashraf, M. Zaman, and M. Ahmed, “An Intelligent Prediction System for Educational Data Mining Based on Ensemble and Filtering approaches,” *Procedia Comput. Sci.*, vol. 167, pp. 1471–1483, 2020, doi: 10.1016/j.procs.2020.03.358.

[29] Z. P. Brodeur, J. D. Herman, and S. Steinschneider, “Bootstrap Aggregation and Cross-Validation Methods to Reduce Overfitting in Reservoir Control Policy Search,” *Water Resour. Res.*, vol. 56, no. 8, Aug. 2020, doi: 10.1029/2020WR027184.

[30] S. Boukir and W. Feng, “Boundary bagging to address training data issues in ensemble classification,” in *2020 25th International Conference on Pattern Recognition (ICPR)*, Jan. 2021, pp. 9975–9981, doi: 10.1109/ICPR48806.2021.9413055.

[31] H. Du and Y. Zhang, “Network anomaly detection based on selective ensemble algorithm,” *J. Supercomput.*, vol. 77, no. 3, pp. 2875–2896, Mar. 2021, doi: 10.1007/s11227-020-03374-z.

[32] P. Melin, J. C. Monica, D. Sanchez, and O. Castillo, “Multiple Ensemble Neural Network Models with Fuzzy Response Aggregation for Predicting COVID-19 Time Series: The Case of Mexico,” *Healthcare*, vol. 8, no. 2, p. 181, Jun. 2020, doi: 10.3390/healthcare8020181.

[33] A. Theissler, M. Thomas, M. Burch, and F. Gerschner, “ConfusionVis: Comparative evaluation and selection of multi-class classifiers based on confusion matrices,” *Knowledge-Based Syst.*, vol. 247, p. 108651, Jul. 2022, doi: 10.1016/j.knosys.2022.108651.