Comparative analysis of classification techniques for leaves and land cover texture

Azri Azrul Azmer [a,1], Norlida Hassan [a,2], Shihab Hamad Khaleefah [b,3], Salama A Mostafa [a,4,*] Azizul Azhar Ramli [a,5],

[a] Faculty of Computer Science & Information Technology, Universiti Tun Hussein Onn Malaysia, Johor, Malaysia
[b] Faculty of Computer Science, Al Maarif University College, Anbar, Iraq.
[1] di170039@siswa.uthm.edu.my; [2] norlida@uthm.edu.my; [3] shi90hab@gmail.com; [4] salama@uthm.edu.my; [5] azizulr@uthm.edu.my
* corresponding author

ARTICLE INFO

Article history
Received March 1, 2021
Revised August 23, 2021
Accepted August 23, 2021
Available online November 30, 2021

Keywords
Data mining
Texture analysis
Random forest
Naive bayes
k-Nearest neighbor

ABSTRACT

The texture is the object's appearance with different surfaces and sizes. It is mainly helpful for different applications, including object recognition, fingerprinting, and surface analysis. The goal of this research is to investigate the best classification models among the Naive Bayes (NB), Random Forest (DF), and k-Nearest Neighbor (k-NN) algorithms in performing texture classification. The algorithms classify the leaves and urban land cover of texture using several evaluation criteria. This research project aims to prove that the accuracy can be used on data of texture that have turned in a group of different types of data target based on the texture's characteristic and find out which classification algorithm has better performance when analyzing texture patterns. The test results show that the NB algorithm has the best overall accuracy of 78.67% for the leaves dataset and 93.60% overall accuracy for the urban land cover dataset.

1. Introduction

Classification is one of the techniques in Data Mining that categories the data collection target. Data Mining is a process of extracting useful information about the data from a large set of raw data [1]. Machine learning is an application of artificial intelligence that can automatically learn and improve from experience that has been trained [2]. The texture is an object's appearance with a different surface, size, characteristic, and texture pattern [3]. Previous studies have applied many data mining approaches in classification to analyze specific or selected data [4][5]. Furthermore, some research tries to improve the data mining technique to boost performance when analyzing the data [6][7]. Besides, several studies use data mining techniques to identify data in a data group (statistics or diagrams) to simplify the data and make it easier to understand [8].

Most studies improve data processing techniques and compare them with previous techniques [9]. The process becomes a reference for improving new (proposed) techniques with better performance and accuracy [10]–[12]. Plants play an essential role in our environment. There will be no earth's ecology [11] to protect plants without plants. This study proposes an automatic classification scheme for plant species based on the shape features in the image of the leaves. Similar research has been carried out by [11]–[13], which classifies the same data, but the difference is in the method and framework.

In addition to implementing classification techniques with three different algorithms based on leaf texture data and urban land cover, this study determine which machine learning has better accuracy and performance during texture data analysis [13] [14]. Classification is a supervised learning approach to

classify new observations. This method determines what data should be categorized by providing a set of sample data classes. It consists of two-phase when constructing a classifier. The training set needs to decide how the parameter should be focused on and combine the different data types into one type in the training phase. The set will be tested by applying the test data with a known target and comparing the training set with selected data. For additional information in the testing, set the result that produced on how long it takes to show the result on each data with the accuracy the data interpret either it has high accuracy or not [15].

The classification process to solve the problem in this study is to use three classification methods, namely Naive Bayes (NB), Random Forest (RF) or Random Decision Forest (RDF), and k-Nearest Neighbor (k-NN). Data texture analysis becomes a reference in the classification process from these methods. Thus, the data texture becomes input in data mining, especially the classification process.

The rest of the paper is organized as follows. Section 2 elucidates the related work on classification algorithms. Based on this research, the explanation of the classification process for obtaining valid results is present in Section 3. Section 4 describes the test and algorithm results, including parameter selection, experimental setup, and comparison of performance and accuracy of previous results. The last section (Section 5) presents a summary and future work.

2. Method

2.1. Random Forest

Random Forest (RF) is a type of ensemble method that is used to predict the average of several independent base models was introduced for classification and regression method for RF framework [16][17]. The ensemble method uses multiple learning algorithms to obtain better predictive performance in classification and regression. One ensemble method used in RF is Bagging (Bootstrap Aggregation). Bagging is one of the techniques that perform in the decision tree used to reduce the variance of a decision tree. The RF model structures it almost like a decision tree model in classification [18]. This method was introduced because each decision tree is constructed by using a random subset of the training data. In the Concept of RF, the data sampling used is random in data training and subset [19]. However, without the independent decision tree, the randomness of the RF method cannot be used. Thus, in order to perform the RF method, creating a decision tree should first be done by using the equation for expected Information, Entropy, and information gain as know decision tree (CART) to create a decision tree [20][21], so the random value of data can be wrong and true. Gini Impurity is one of the measurements for the RF. The advantage of RF is that it can handle missing data values and maintain the accuracy for the missing data. However, it has a disadvantage where it would not give the precise value when using a regression model. That is why the RF uses random with multiple learning algorithms to get better performance and accuracy from the decision tree method [22]. Concept of RF, the data sampling used is random in data training and subset. However, without the independent decision tree, the randomness of the RF method cannot be used. So in order, perform the RF method, creating a decision tree should do first by using the equation for expected information (Fig. 1).

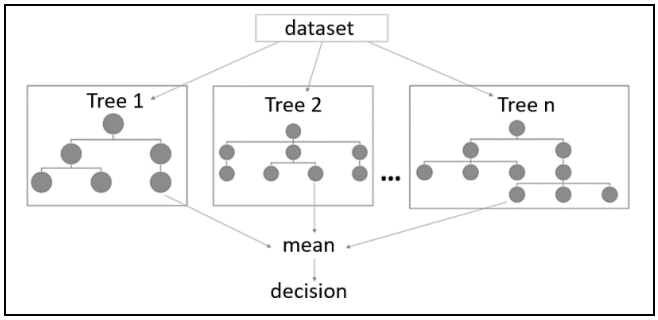

Fig. 1. The graphical model of the RF

2.2. Naïve Bayes

The Naive Bayes (NB) is a probabilistic algorithm that performs classification tasks based on the Bayes theorem that Thomas Bayes introduced in 1702. Bayes theorem is like assumption was made if a feature or predictor was independent that does not affect others in the dataset [23]. NB has three different types of NB, like Multinomial, that mainly use the document classification problem. Bernoulli usually uses in prediction in Boolean values like true and false and 0 and 1 and Gaussian Naive use in prediction by using continuous value datasets. Using NB has their advantage and disadvantage, like if the prediction is accurate, then the performance becomes better compared to other like logistic regression with less training data. It is also straightforward to implement, but it has a high chance to lower data accuracy [24]. NB is a supervised machine learning algorithm usually used for classification that can assume the presence of a particular feature in a class unrelated to the presence feature (Fig. 2) [25].

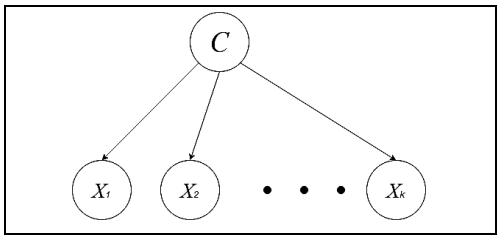

Fig. 2. The graphical model of the NB

2.3. k-Nearest Neighbor

k-Nearest Neighbor (k-NN) is a supervised classification that can classify non-attribute by assigning them to a similar attribute in the class, as shown in Fig. 3 [26]. Based on the articles related to k-NN, the probability of error of simple classification rule is bounded with the Bayes minimum probability of error is better that make the most impact paper in pattern recognition and texture classification applications such as the document authentication texture features [27].

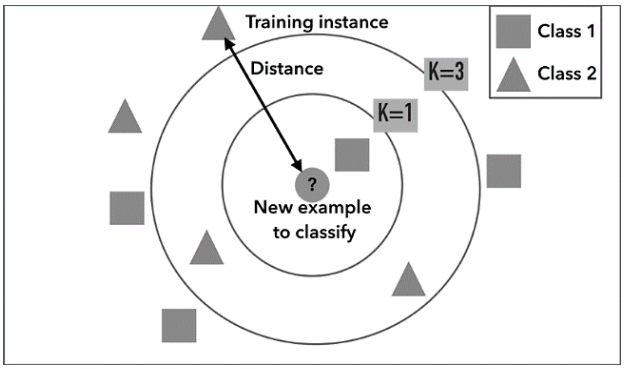

Fig. 3. The graphical model of the k-NN

The k-NN algorithm is one of the supervised machine learning algorithms that can solve both classification and regression problems. It is straightforward to implement and understand but has the significant drawback of becoming significantly slow when using a large dataset. This model consists of a few of the equations that estimated the distance between the variable of the dataset. In the distance, function consists of three types like Euclidean, widely used in every article and journal, Manhattan and Minkowski. For additional information, there is another distance measure where it uses standardized distance. This standardized distance happens when there is a case wherein a training set consists of a combination variable like numerical and categorical. In k-NN, there are pros and cons when using this model as this model is straightforward to implement and understand where k-NN is known as Lazy

Leaner because this model does not have a training period that makes this model algorithm has faster training performance than others like SVM and linear regression. However, this model has a significant problem with slower performance when handling a large dataset. Another problem with k-NN this model is sensitive to noisy data and missing values that need to be input personally [28].

2.4. Research flow

Fig. 4 shows the entire research framework on how the result will be produced from the leaves and urban land datasets, including the accuracy.

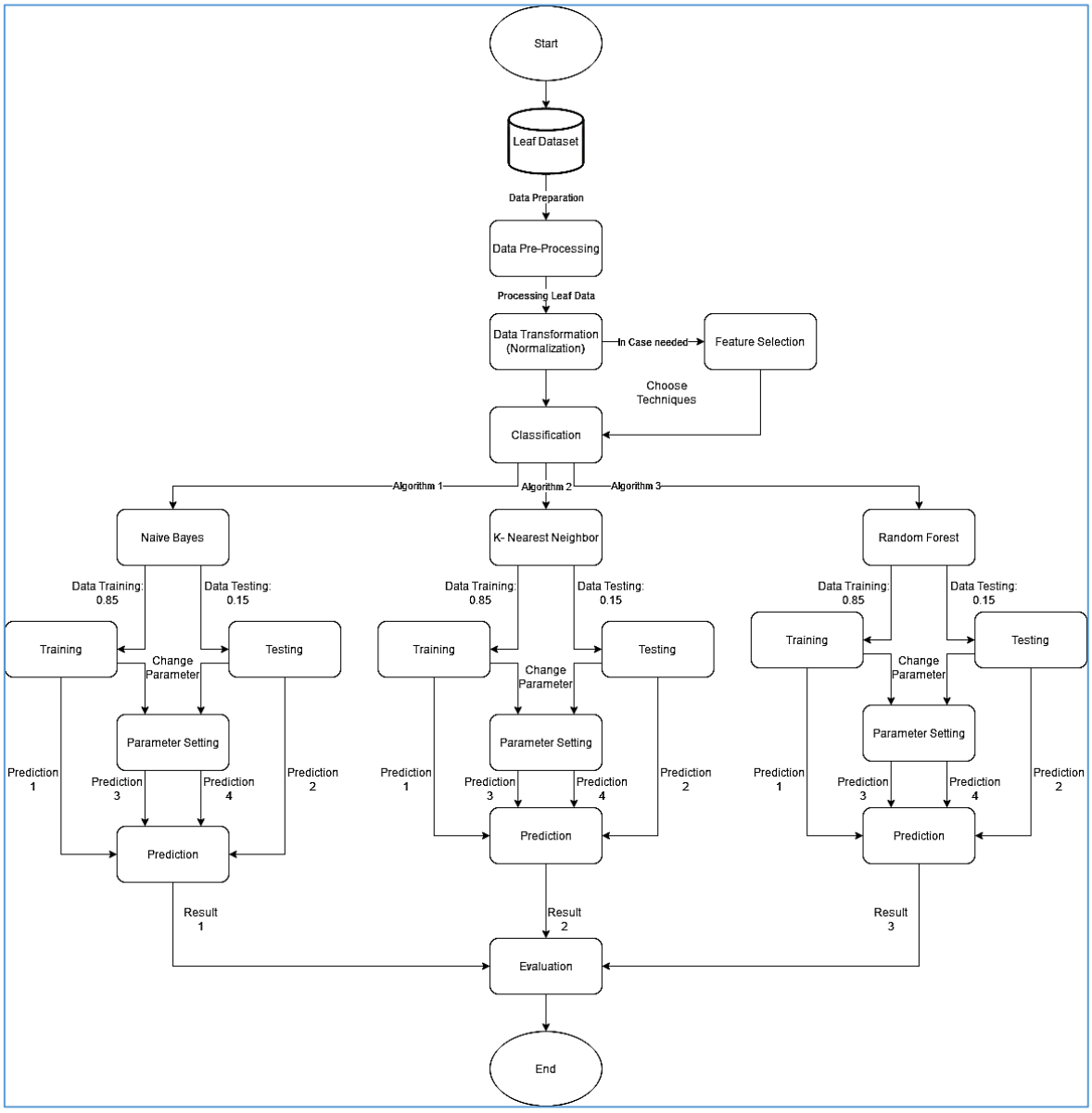

Fig. 4. A research framework for the classification of leaves and urban land cover datasets

2.4.1. Data Selection

Data used is the leaves dataset and urban land cover taken from the Machine Learning Repository (UCI). The leaves dataset consists of shape and texture features extracted from digital images of leaf specimens originating from 40 different plant species. With 40 plant species, this dataset has 340 data with 16 attributes. The urban land cover dataset is a high-resolution aerial image of 9 urban land cover types with parameters, such as multi-scale spectral, size, shape, and texture information. However, this

dataset has a total of 507 data with 22 attributes. Each of the data was in numerical form in continuous value that was suitable by using a classification in predicting the accuracy of each technique.

2.4.2. Data Pre-processing

In the preprocessing data phase, some of the methods are used on the leaf's dataset and urban land cover dataset before data simulation using the selected algorithms of NB, RF, and k-NN. The primary data processing methods are cleaning, normalization, and reduction. Data cleaning will process the leaf datasets if any missing value needs to be worked. However, in this case, Leaf data originally does not have any missing value in the dataset, so data cleaning will not perform in this research project. The urban land cover dataset consists of a problem where there is some data redundant and some missing values in the dataset. In this case, to handle duplicated and missing values, the related raw data will be removed to avoid unprocessed data during the training and testing phases. For data, the reduction is a process where a large dataset will be removed to get results faster because the fewer dataset is, the faster the result will be produced. However, in this research project, data reduction will be performed on the urban land cover dataset only because there is duplicate data, and this data reduction is already performed during the data cleaning process that makes the final total of data into 507.

Data normalization is a process of transforming the data wherein this research project the leaves datasets and urban land dataset need to transform it into continuous value where the data with a value between 0 and 1 [29]. In this research project, MinMax normalization will ensure that all the data in the dataset has a normalized value of each row of data (1).

$$MinMax = \frac{(v - Min\ x)}{Max\ x - Min\ x}(newMax - newMin) + newMin \quad (1)$$

Min refers to the lowest value of the data in the leaves and urban land datasets. Max refers to the highest value of the data in the leaves and urban land dataset, V is the pick value of the row on each attribute of leaves dataset and urban land dataset, newMax sets the maximum value to 1. newMin sets the minimum value to 0.

2.4.3. Performance Measurements

For the performance measurements to find the accuracy, sensitivity, and specificity of the techniques, the research paper was [30]. Accuracy is a calculated correct prediction divided into a total number of data in the dataset. Sensitivity calculates all the correct predictions that have been divided into a total number of correct predictions. Specificity calculates all the correct predictions divided into a total number of incorrect predictions. The best value for accuracy, sensitivity, and specificity was 1.0 or 100%, and the worst value was 0.0 or 0%. This equation will be used during the training and testing model to find each algorithm's accuracy, sensitivity, and specificity. The primary performance measure is evaluated in accuracy, precision, and recall from the classification confusion matrix [31], [32]. The measures are computed by using (2)-(5).

Accuracy is the total number of samples correctly classified to the total number of samples classified. The formula for calculating accuracy is shown in (2). The number of samples is classified as positive divided by the total sample in the testing set positive category. The formula for calculating recall is shown in (3).

$$Accuracy = \frac{TP+TN}{TP+TN+FN+FP} = \frac{TP+TN}{P+N} \quad (2)$$

$$Sensitivity = Recall = \frac{TP}{TP+FN} = \frac{TP}{P} \quad (3)$$

T is the number of samples is categorized positively classed correctly divided by total samples are classified as positive samples. The formula for calculating precision is shown in (4)

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

The characteristics of the test and measures the proportion of negatives that are correctly identified. The population does not affect the results. The formula for calculating specificity is shown in (5).

$$Specificity = \frac{TN}{TN+FP} \quad (5)$$

3. Results and Discussion

3.1. Parameters Selection and Experimental Setting

Parameter selection processes choose the selected parameter for each technique algorithm used during each of the data training. Each technique will be chosen based on the effect of manipulation on a specific parameter. For RF techniques, the parameter used was *mtry* and *ntree*. For k-NN, *k* will be used as a parameter, and NB uses the feature as a chosen parameter tuned in every model to increase the accuracy of the performance. Table 1, Table 2, and Table 3 show the selected parameters of each classification model with the description of each parameter.

Table 1. The selected parameter for each technique in the leaves dataset

No.	Technique	Default Parameter	Parameter Selected (Testing)
1	RF	mtry : 8 ntree : 500	mtry :1-20 ntree :300 - 2000
2	k-NN	K=17 K=18	K= 1-50
3	NB	Feature: Class..species.	Feature : All feature

Table 2. The selected parameter for each technique in the Land Urban dataset

No.	Technique	Default Parameter	Parameter Selected (Testing)
1	RF	mtry : 8 ntree : 500	mtry :range(1-20) ntree :range(300 - 2500)
2	k-NN	K=23 K=24	K= 1-50
3	NB	Feature: classes.	Feature : All feature

Table 3. The description of each selected parameter for each technique

No.	Technique	Parameter	Parameter Description
1	RF	Mtry Ntree	Mtry: the number of variables that will randomly sample used as candidates at each tree split. Ntree: the number tree to grow.
2	k-NN	k	k: the number of the nearest neighbor of the model will be considered based on the length of the data.
3	NB	Feature	Feature: the attribute of the dataset.

Each model's parameters have been trying to manipulate the value from the original parameter and the list parameter in the tables. The parameters tuning shows an improvement in the accuracy of each model. For this experiment testing, Rstudio has to be used as the platform for each data's training, and testing model with a different technique that will be used is the RF, k-NN, and NB to make this research successful produce the result. As mentioned in the training and testing model discussed just now, this experiment requires creating these two models for each algorithm. Using these two models in the training model ensures that the algorithm can be trained using the leaves and the urban land dataset. In addition, each of these algorithms cannot be applied directly if the data do not match. This algorithm needs to be trained first before the actual test. The other model is the testing model, and the testing model is a model where the performing the real test to get the output which means accuracy and performance. Each algorithm validation and evaluation was applied to see the expected outcome meets

the project objectives. Table 4 and Table 5 show the information related to Rstudio and the initial set of splitting the data based on 0.75 of training and 0.25 of testing for each dataset used.

Table 4. Data Information

No	Data	Number of data	Number of the attribute
1	Leaves dataset	340	16
2	Urban land dataset	507	22

Table 5. Data Split

No	Data	Training (0.75)			Test (0.25)		
		RF	*k-NN*	*NB*	*RF*	*K-NN*	*NB*
1	leaves dataset	259	255	265	81	85	75
2	Urban land dataset	376	380	382	131	127	125

3.2. Experiment Results

The RF in Rstudio required an extension package as training and testing for the RF algorithm. At the initial phase of the experiment, the leaves dataset and urban land cover were normalized to optimize the data. All the datasets have been separated into two different data models for the training and testing models and then processed the data by using 10-fold cross-validation to increase the effectiveness of the models. Table 6 and Table 7 show the final result of the testing models with the selected parameter change on both datasets.

Table 6. The result of testing the RF model with the selected parameters of the leaves dataset

N=81	r		Measure	
Prediction (predict)			*Precision (PPV)*	
			72.84%	
Positive	59	22	*NPV*	
	22	0	0%	
Measure	*Sensitivity*	*Specificity*	*Accuracy*	*Error rate*
	72.84%	0%	57.28%	42.72%
			Overall Accuracy	*Error rate*
			72.84%	27.16%

Table 7. The result of testing the RF model with the selected parameters of the urban land cover dataset

N=131	r		Measure	
Prediction (predict)			*Precision (PPV)*	
			81.68%	
Positive	107	24	*NPV*	
	24	0	0%	
Measure	*Sensitivity*	*Specificity*	*Accuracy*	*Error rate*
	81.68%	0%	69.03%	30.97%
			Overall Accuracy	*Error rate*
			81.68%	18.32%

Different parameter settings on the RF algorithm show 72.84 % of overall accuracy with 27.16% error rate on leaves dataset and for urban land cover also gain 81.68% of overall accuracy with 18.32% error rate on the final testing model. The k-NN in Rstudio also required an extension package used as

training and testing for the k-NN algorithm. This experiment has been using the same method in initial phase data normalization. Table 8 and Table 9 show the final result of the experiment by the k-NN algorithm on two different datasets.

Table 8. The result of testing the k-NN model with selected parameters of the leaves dataset

N=131	r		Measure	
Prediction (predict)			*Precision (PPV)*	
			44.71%	
Positive	38	47	*NPV*	
	47	0	0%	
Measure	*Sensitivity*	*Specificity*	*Accuracy*	*Error rate*
	44.71%	0%	28.79%	71.21%
			Overall Accuracy	*Error rate*
			44.71%	55.29%

Table 9. The result of testing the k-NN model with selected parameters of the urban land cover dataset

N=127	r		Measure	
Prediction (predict)			*Precision (PPV)*	
			66.14%	
Positive	84	43	*NPV*	
	43	0	0%	
Measure	*Sensitivity*	*Specificity*	*Accuracy*	*Error rate*
	66.14%	0%	49.41%	50.59%
			Overall Accuracy	*Error rate*
			66.14%	33.86%

Changing one parameter setting on the k-NN algorithm shows 44.71% of overall accuracy with a 55.29% error rate on the leaves dataset. Moreover, urban land cover gains 66.14% overall accuracy with a 33.86% final testing model error rate. The NB in Rstudio requires the same extension package as the k-NN experiment before training and testing the k-NN algorithm. This experiment has also used the same method in initial phase data normalization 10-fold cross-validation on the previous experiment.

Table 10 and Table 11 show the final result of the experiment by the k-NN algorithm on two different datasets. Changing one parameter setting on the k-NN algorithm shows 78.67% of overall accuracy with a 21.33% error rate on the leaves dataset. Moreover, urban land cover gains 93.60% overall accuracy with a 6.40% final testing model error rate.

Table 10. The result of testing the NB model with the selected parameters of the leaves dataset

N=75	r		Measure	
Prediction (predict)			*Precision (PPV)*	
			78.67%	
Positive	59	16	*NPV*	
	16	0	0%	
Measure	*Sensitivity*	*Specificity*	*Accuracy*	*Error rate*
	78.67%	0%	64.84%	35.16%
			Overall Accuracy	*Error rate*
			78.67%	21.33%

Table 11. The result of testing the NB model with selected parameters of the urban land cover dataset

N=125	r		Measure	
Prediction (predict)			*Precision (PPV)*	
			93.60%	
Positive	117	8	*NPV*	
	8	0	0%	
Measure	*Sensitivity*	*Specificity*	*Accuracy*	*Error rate*
	93.60%	0%	87.97%	12.03%
			Overall Accuracy	*Error rate*
			93.60%	6.40%

Evaluation of the performance of each algorithm using the concept of multiclass on each dataset. The summary results of all experiments carried out in the test are shown in Table 12. The calculation model uses each algorithm and object's accuracy and error rate (leaves and urban land cover).

Table 12. Summary of overall experimental and result

Algorithm	Leaves dataset		Urban land cover dataset	
	Experiment Measure			
	Overall Accuracy	*Error rate*	*Overall Accuracy*	*Error rate*
RF	72.84%	27.16%	81.68%	18.32%
k-NN	44.71%	55.29%	66.14%	33.86%
NB	78.67%	21.33%	93.60%	6.40%

Table 12 shows the overall accuracy performance with an error rate of each algorithm in RF, k-NN, and NB on two datasets in texture in leaves dataset and urban land cover dataset. NB algorithm has the best overall accuracy where the NB has 78.67% overall accuracy performance with 21.33% error rate compared to RF and k-NN algorithm on leaves dataset and also urban land cover where NB has 93.60% overall accuracy with 6.40% error rate compared to the RF and k-NN algorithm

4. Conclusion

Based on the research aims, the test and evaluation might increase the accuracy as it can deduce which algorithm best suits texture analysis. The selected algorithms are Naive Bayes (BN), Random Forest (RF), and k- Nearest Neighbor (k-NN), and the selected datasets are leaves and also urban land cover. The texture dataset can be classified based on the type and pattern of texture. The type and pattern of the texture significantly affect the performance accuracy of the classification algorithms. The tests show that the NB algorithm has the best overall accuracies of 78.67% and 93.60% for respective leaves and urban land cover datasets compared to the RF and k-NN results. Hence, algorithms with independent features analysis capability are found to be more sensitive to texture features along. Including ensemble techniques might increase the performance and accuracy of the ML algorithms and could be covered in our future work.

Acknowledgment

This paper is supported by Research Fund E15501, Research Management Centre, Universiti Tun Hussein Onn Malaysia (UTHM).

Declarations

Author contribution. All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.

Funding statement. This paper is supported by Research Fund E15501, Research Management Centre, Universiti Tun Hussein Onn Malaysia (UTHM).
Conflict of interest. The authors declare no conflict of interest.
Additional information. No additional information is available for this paper.

References

[1] P. Sunhare, R. R. Chowdhary, and M. K. Chattopadhyay, "Internet of things and data mining: An application oriented survey," *J. King Saud Univ. - Comput. Inf. Sci.*, Jul. 2020, doi: 10.1016/j.jksuci.2020.07.002.

[2] S. L. Goldenberg, G. Nir, and S. E. Salcudean, "A new era: artificial intelligence and machine learning in prostate cancer," *Nat. Rev. Urol.*, vol. 16, no. 7, pp. 391–403, Jul. 2019, doi: 10.1038/s41585-019-0193-3.

[3] S. H. Khaleefah, S. A. Mostafa, A. Mustapha, and M. F. Nasrudin, "The ideal effect of Gabor filters and Uniform Local Binary Pattern combinations on deformed scanned paper images," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 33, no. 10, pp. 1219–1230, Dec. 2021, doi: 10.1016/j.jksuci.2019.07.012.

[4] A. Tharwat, "Classification assessment methods," *Appl. Comput. Informatics*, vol. ahead-of-p, no. ahead-of-print, Aug. 2020, doi: 10.1016/j.aci.2018.08.003.

[5] S. H. Khaleefah and M. F. Nasrudin, "Identification of printing paper based on texture using gabor filters and local binary patterns," *J. Theor. Appl. Inf. Technol.*, vol. 86, pp. 279–289, 2016. Available Semantic Scholar.

[6] H. Hartono and E. Ongko, "Hybrid approach redefinition with progressive boosting for class imbalance problem," *Sci. Inf. Technol. Lett.*, vol. 1, no. 1, pp. 40–51, Apr. 2020, doi: 10.31763/sitech.v1i1.34.

[7] N. Saravana Kumar, K. Hariprasath, N. Kaviyavarshini, and A. Kavinya, "A study on forecasting bigmart sales using optimized machine learning techniques," *Sci. Inf. Technol. Lett.*, vol. 1, no. 2, pp. 52–59, Nov. 2020, doi: 10.31763/sitech.v1i2.167.

[8] P. F. B. Silva, A. R. S. Marçal, and R. M. A. da Silva, "Evaluation of Features for Leaf Discrimination," *Kamel M., Campilho A. Image Anal. Recognition. ICIAR 2013. Lect. Notes Comput. Sci.*, pp. 197–204, 2013, doi: 10.1007/978-3-642-39094-4_23.

[9] H. Ait Issad, R. Aoudjit, and J. J. P. C. Rodrigues, "A comprehensive review of Data Mining techniques in smart agriculture," *Eng. Agric. Environ. Food*, vol. 12, no. 4, pp. 511–525, Oct. 2019, doi: 10.1016/j.eaef.2019.11.003.

[10] R. Sahani, Shatabdinalini, C. Rout, J. Chandrakanta Badajena, A. K. Jena, and H. Das, "Classification of Intrusion Detection Using Data Mining Techniques," *Pattnaik P., Rautaray S., Das H., Nayak J. Prog. Comput. Anal. Networking. Adv. Intell. Syst. Comput. vol 710. Springer, Singapore.*, pp. 753–764, 2018, doi: 10.1007/978-981-10-7871-2_72.

[11] N. Tomasevic, N. Gvozdenovic, and S. Vranes, "An overview and comparison of supervised data mining techniques for student exam performance prediction," *Comput. Educ.*, vol. 143, p. 103676, Jan. 2020, doi: 10.1016/j.compedu.2019.103676.

[12] A. P. Gopi, R. N. S. Jyothi, V. L. Narayana, and K. S. Sandeep, "Classification of tweets data based on polarity using improved RBF kernel of SVM," *Int. J. Inf. Technol.*, Jan. 2020, doi: 10.1007/s41870-019-00409-4.

[13] S. A. Mostafa *et al.*, "Examining multiple feature evaluation and classification methods for improving the diagnosis of Parkinson's disease," *Cogn. Syst. Res.*, vol. 54, pp. 90–99, May 2019, doi: 10.1016/j.cogsys.2018.12.004.

[14] B. A. Johnson, "Mapping urban land cover using multi-scale and spatial autocorrelation information in high resolution imagery," 2012. Available : Google Scholar.

[15] S. Mohan and K. Venkatachalapathy, "Wood Knot Classification using Bagging," *Int. J. Comput. Appl.*, vol. 51, no. 18, pp. 50–53, Aug. 2012, doi: 10.5120/8146-1937.

[16] S. A. Mostafa, A. Mustapha, M. A. Mohammed, M. S. Ahmad, and M. A. Mahmoud, "A fuzzy logic control in adjustable autonomy of a multi-agent system for an automated elderly movement monitoring

application," *Int. J. Med. Inform.*, vol. 112, pp. 173–184, Apr. 2018, doi: 10.1016/j.ijmedinf.2018.02.001.

[17] M. Denil, D. Matheson, and N. De Freitas, "Narrowing the Gap: Random Forests In Theory and In Practice," *Proc. 31st Int. Conf. Mach. Learn.*, vol. 32, no. 1, pp. 665–673, 2014, [Online]. Available: https://proceedings.mlr.press/v32/denil14.html.

[18] G. T. P. Kumari, "A Study of Bagging and Boosting approaches to develop meta-classifier," *Eng. Sci. Technol. An Int. J.*, vol. 2, no. 5, pp. 850–855, 2012. Available: Semantic Scholar.

[19] J. Dou *et al.*, "Assessment of advanced random forest and decision tree algorithms for modeling rainfall-induced landslide susceptibility in the Izu-Oshima Volcanic Island, Japan," *Sci. Total Environ.*, vol. 662, pp. 332–346, Apr. 2019, doi: 10.1016/j.scitotenv.2019.01.221.

[20] T. Wang *et al.*, "Random Forest-Bayesian Optimization for Product Quality Prediction With Large-Scale Dimensions in Process Industrial Cyber–Physical Systems," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 8641–8653, Sep. 2020, doi: 10.1109/JIOT.2020.2992811.

[21] R. Pal, S. Kapali, and S. Trivedi, "A Study on Credit Scoring Models with different Feature Selection and Machine Learning Approaches," *SSRN Electron. J.*, 2020, doi: 10.2139/ssrn.3743552.

[22] S. Touzani, J. Granderson, and S. Fernandes, "Gradient boosting machine for modeling the energy consumption of commercial buildings," *Energy Build.*, vol. 158, pp. 1533–1543, Jan. 2018, doi: 10.1016/j.enbuild.2017.11.039.

[23] G. Meena and R. R. Choudhary, "A review paper on IDS classification using KDD 99 and NSL KDD dataset in WEKA," in *2017 International Conference on Computer, Communications and Electronics (Comptelix)*, Jul. 2017, pp. 553–558, doi: 10.1109/COMPTELIX.2017.8004032.

[24] M. Swathy and K. Saruladha, "A comparative study of classification and prediction of Cardio-Vascular Diseases (CVD) using Machine Learning and Deep Learning techniques," *ICT Express*, Sep. 2021, doi: 10.1016/j.icte.2021.08.021.

[25] A. Yasar, I. Saritas, M. A. Sahman, and A. O. Dundar, "Classification of Leaf Type Using Artificial Neural Networks," *Int. J. Intell. Syst. Appl. Eng.*, vol. 3, no. 4, p. 136, Dec. 2015, doi: 10.18201/ijisae.49279.

[26] A. Music and S. Gagula-Palalic, "Classification of Leaf Type Using Multilayer Perceptron, Naive Bayes and Support Vector Machine Classifiers," *Southeast Eur. J. Soft Comput.*, vol. 5, no. 2, Oct. 2016, doi: 10.21533/scjournal.v5i2.119.

[27] S. H. Khaleefah, M. F. Nasrudin, and S. A. Mostafa, "Fingerprinting of deformed paper images acquired by scanners," *2015 IEEE Student Conf. Res. Dev.*, pp. 393–397, Dec. 2015, doi: 10.1109/SCORED.2015.7449363.

[28] G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, "KNN Model-Based Approach in Classification," *Meersman R., Tari Z., Schmidt D.C. Move to Meaningful Internet Syst. 2003 CoopIS, DOA, ODBASE. OTM 2003. Lect. Notes Comput. Sci. vol 2888. Springer, Berlin, Heidelb.*, pp. 986–996, 2003, doi: 10.1007/978-3-540-39964-3_62.

[29] S. G. K. Patro and K. K. Sahu, "Normalization: A Preprocessing Stage," Mar. 2015, [Online]. Available: http://arxiv.org/abs/1503.06462.

[30] S. Visa, B. Ramsay, A. L. Ralescu, and E. Van Der Knaap, "Confusion matrix-based feature selection.," *MAICS*, vol. 710, pp. 120–127, 2011. Available: Google Scholar.

[31] S. A. Mostafa, A. Mustapha, S. H. Khaleefah, M. S. Ahmad, and M. A. Mohammed, "Evaluating the Performance of Three Classification Methods in Diagnosis of Parkinson's Disease," *Ghazali R., Deris M., Nawi N., Abawajy J. Recent Adv. Soft Comput. Data Mining. SCDM 2018. Adv. Intell. Syst. Comput. vol 700. Springer, Cham*, pp. 43–52, 2018, doi: 10.1007/978-3-319-72550-5_5.

[32] R. M. De Moraes and L. D. S. Machado, "Online Training Assessment in Virtual Reality Simulators Based on Gaussian Naive Bayes," *Comput. Intell. Decis. Control*, pp. 1147–1152, Aug. 2008, doi: 10.1142/9789812799470_0188.