Nat. Volatiles & Essent. Oils, 2021; 8(5): 8516- 8530

NVEO
Natural Volatiles &
Essential Oils

Classification Of Spam Detection Using Naive Bayes Algorithm Over K-Nearest Neighbors Algorithm Based On Accuracy

A. Gayathri[1], J. Aswini[2] and A. Revathi[3]

[1]Associate Professor, Department of Computer Science and Engineering, Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences (SIMATS), Chennai, India
Email: gayathribala.sse@saveetha.com

[2]Professor, Department of Computer Science and Engineering, Sree Vidyanikethan Engineering College, Tirupati, India
Email: aswini.jayaraman@gmail.com

[3]Assistant Professor, Department of CSE, School of computing, SRM Institute of Science and Technology, Kattangulathur
Email: revathia1@srmist.edu.in

ABSTRACT

AIM: To predict the accuracy percentage of Short Message Services(SMS) spam detection using machine learning classifiers. **Materials and Methods** :Two ensemble learning algorithms named naive bayes and k-nearest neighbors are applied to data.The algorithms have been implemented and tested over a dataset which consists of 5574 records. Ensemble learning methods combined several models trained with a given learning algorithm to improve accuracy. **Results and Discussion:** After performing the experiment as result shows mean accuracy of 88.05 % by using naive bayes algorithm and compared k-nearest neighbor algorithm mean accuracy is 58.04% for SMS spam detection.There is a statistical significant difference in accuracy for two algorithms is $p<0.05$ by performing independent samples t-tests. **Conclusion:** This paper is intended to implement Innovative machine learning classifiers for prediction of SMS spam detection.The comparison results shows that the naive bayes algorithm has appeared to be better performance than k-nearest neighbor algorithm.

Keywords: Spam words, Naive Bayes Algorithm ,K-Nearest Neighbor Algorithm, Innovative Methods, Innovative Machine Learning, Classifiers.

INTRODUCTION

The primary initiation of the study is performed to implement a spam detector to detect the spam words.Mobile phone spam also known as (unsolicited messages, especially advertising), directed at the text messaging or other communications services of mobile phones or smartphones(Patterson, n.d.). Fighting SMS spam is complicated by several factors (compared to Internet email), including the lower rate of SMS spam, which has allowed many users and service providers to ignore the issue, and the limited availability of mobile phone spam-filtering software.(Arifin, Shaufiah, and Bijaksana 2016). In the paper, implemented to analyze different Innovative methods to identify spam/ham messages. This paper uses a different approach to establish relation between the text and the category, based on size of message, word count, special keywords, using term-frequency inverse document-frequency (tf-idf) transform.(Abbashi et al. 2020)

There are nearly 22 articles published in science direct and 26 articles published on the web of science related to spam detection. In paper(Kołcz 2017) a data mining procedure is performed on messages to collect the spam words and convert it into a dataset which can be used for analysis and is cited about 324 times as reference for research. There are many other machine learning classifiers implemented earlier to detect spam words published or shared over messages that have minimal accuracy(Trivedi 2016). This paper proposes implementing a machine learning classifier which can provide a better accuracy for spam detection published over messages than the previously implemented classifiers(Gupta et al. 2018). A spam detector is implemented which is used to detect

the spam words published on messages using multiple sources and various classes , proposed in paper(Khanday 2019).The classifiers which are used in previous proposed papers have less accuracy rate so here implemented naive bayes algorithm classifiers to give improved accuracy and comparing it with k-nearest neighbor algorithm. It is a supervised machine learning algorithm which is similar to the classifiers implemented in paper like random forest, Naive Bayes. This paper (Aydogan and Karci 2018)is best for future researchers who are interested in spam detection as a reference, a data mining procedure is performed on messages to collect the spam words and convert it into a dataset which can be used for analysis.

The methods which are used before have less accuracy rate, are less reliable and not much effective in prediction of spam detection. The main aim of the study is proposed to perform classification of spam detection by implementing spam detectors using machine learning classifiers like naive bayes algorithm and k-nearest neighbor algorithm and comparing their performance.

MATERIALS AND METHODS

This proposed work involves two supervised learning algorithms, Naive Bayes algorithm and K-Nearest Neighbors algorithm. It performs two iterations for spam detection using these two algorithms.The pre-test analysis was done using clincalc.com(Kane, Phar, and BCPS n.d.) by keeping g-power at 81%.

"Spam dataset" is used in this paper. The dataset was collected from the open source("Website" n.d.). This is a dataset which consists of data related to the spam words. The dataset contains file names such as "spam.csv" and attributes such as "V1", "V2". In this work, only the text attribute is considered for analysis and classification of spam words.

Naive bayes algorithm :

Naive Bayes algorithm is a classification technique based on Bayes theorem with an assumption of independence among predictors. This model is easy to build and also used for large datasets .The only drawback of naive bayes algorithms is to assume all factors as dependent on each variable. It is mainly based on the theorem formulated by Bayes's given in equation (1):

$$P(A|B)=P(B|A).P(A)/P(B) \quad \ldots\ldots (1)$$

Where,

A, B = events

P(A|B) = probability of A given B is true

P(B|A) = probability of B given A is true

P(A), P(B) = the independent probabilities of A and B

Pseudocode: Naive Bayes Algorithm:

Input: Training Set

Output: Classifiers trained accuracy

Step 1:Read the trained dataset into the classifier.

Step 2:Calculate the mean and standard deviation for predictions.

Step 3:Repeat

Calculate gauss density for each iteration

Until probability of fake political news texts are calculated

Step 4:Define class

def Multinomial()

if(condition satisfy)

return accuracy

else

return to previous step

end

Step 5:Predicted Accuracy

K-Nearest Neighbors algorithm :

K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique and it can be used for Regression as well as for Classification. But mostly it is used for the Classification problems. K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data. It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset**.**

pseudocode for k-nearest neighbor algorithm:

Input: Training Set

Output: Classifiers trained accuracy

Step 1:Calculate "d(x, xi)" i =1, 2, ….., n; where d denotes the Euclidean distance between the points.

Step 2:Arrange the calculated n Euclidean distances in non-decreasing order.

Step 3:Let k be a +ve integer, take the first k distances from this sorted list.

Step 4:Find those k-points corresponding to these k-distances.

Step 5:Let ki denotes the number of points belonging to the ith class among k points i.e. $k \geq 0$

Step 6:If ki >kj $\forall$ i ≠ j then put x in class i.

end

Step 7:Predicted Accuracy

Decision tree Algorithm

Decision tree is a supervised learning algorithm. It is used for both regression and classification. The goal of using a decision tree is to create a training model that can be used to predict the class or value of the target variable by learning simple decision rules developed from training data.

Pseudocode: Decision Tree Algorithm

Input: Trained dataset

Output: Classifier trained accuracy

Step 1:Read the training dataset into the classifier

Step 2:Define a class dtree

Class dtree

Step 3:Get all the required stuff from previous inputs

Step 4:Define another class to test the attribute

def evaluate(test attribute)

if(end iteration is leaf)

return accuracy

else

return children[test attribute].evaluate(test attribute)

end

Step 5:Classifiers predicted accuracy.

Random Forest Algorithm

Random Forest algorithm is a supervised learning algorithm which is used for both regression and classification. A Random Forest is a meta estimator that fits a number of Decision Tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is controlled with the max_samples parameter if bootstrap=True (default), otherwise the whole dataset is used to build each tree. It is an improved version of the Decision Tree.

Pseudocode for Random Forest Algorithm

Input : Training dataset

Output: Classifier tested accuracy

Step 1: Read the training dataset as input

Step 2: Randomly select 'k' samples from total 'm' samples

k<<m

Step 3: Among 'k' tokens calculate the node 'd' using best split

Step 4:Split the node into child nodes

Step 5:Repeat 1 to 3 steps until 'l' number of samples reached

Step 6:Build the forest

Step 7: Predict the value using predict feature

prediction=model.predict(parameters, "")

Step 8: Calculate vote for each predicted value

Step 9: Get the final predicted accuracy

Experiment setup:

The platform used to evaluate the machine learning algorithms was jupyter lab. The hardware configurations were an intel core i5 processor with a RAM size of 4 GB . The System type used was a 64-bit OS,X^$ based processor with HDD of 917 GB. The operating system used was windows and the tool used was jupyter lab with python programming language.

The dataset spam is collected. Data preprocessing has to be done. Data cleaning like removing the duplication of text, incorrect format and unnecessary data from the dataset("Cleaning Text Data" 2018). It also requires concatenating and shuffling need to be done.

Data exploration shows the contents present in the dataset. Convert the dataset that it contains only the data needed for the classifier. Split the dataset into a training set and testing set. Now implement the machine learning classifier and use the training dataset to train the classifier. After training the classifier uses a testing dataset to test the trained classifier to get the predicted accuracy from the classifier.

This paper uses the SPSS tool to perform the statistical calculations for the results and verified results obtained from classifiers for various test sizes. The text part in the training dataset is an independent variable whereas the text part in the testing dataset is dependent on the training dataset.

RESULTS

Table 1 shows to observe the accuracy of the naive bayes algorithm is approximately (88%) and k-nearest neighbor algorithm is approximately (58%). The accuracy varies for different sizes in decimals

From Table 2, it is observed that mean accuracy and standard deviation for naive bayes algorithm is 88.04 and 0.00707. For k-nearest neighbors the algorithm is 58.045 and 0.0707.

From Table 3, it is observed the comparison of accuracy for spam detection classification using naive bayes algorithm and k- nearest neighbor algorithm.

In performing statistical analysis, the naive bayes algorithm obtained 0.007 standard deviation with 0.050 standard error while k-nearest neighbors obtained 0.070 standard deviation with 0.0500 standard error (Table 2). The significance value smaller than 0.001 showed that our hypothesis holds good. With respect to changes in the input values attributes in independent variables are profile, source, proofs and the corresponding output for dependent variables are internal user, external user.

Fig 2 shows the bar chart representing the comparison of mean accuracy of naive bayes algorithm and k-nearest neighbor algorithm. Naïve bayes appear to produce more consistent results with minimal standard deviation.

Table 1: Accuracy performance measure (Naive Bayes Algorithm, K-Nearest Neighbor Algorithm)

Test Size	**0.1**	**0.2**
Naive Bayes Algorithm	88.05	88.06
K-nearest neighbor Algorithm	58.04	58.05

Table 2: Group Statistics (Mean of naive bayes algorithm is 88.04% is more than K-Nearest Neighbor algorithm 58.45% and Error mean for naive bayes algorithm is 0.050 and for K-Nearest Neighbor is 0.92)

	NBA,KNN	**N**	**Mean**	**Std. Deviation**	**Std. Mean Error**
Accuracy	NBA	2	88.04	0.00707	0.050
	KNN	2	58.45	0.07071	0.0500

Table 3: Independent Samples Test (naive bayes algorithm appears to perform significantly better than k-nearest neighbors algorithm)

		Levene's Test for Equality of Variances	**Levene's Test for Equality of Variances**	**T-test for Equality of Means**	**T-test for Equality of Means**	**T-test for Equality of Means**
		F	**Sig.**	**t**	**df**	**Sig. (2-tailed)**
Accuracy	**Equal Variances assumed**	3.246	0.025	588.963	2	0.000

	Equal Variances not assumed	-	-	588.963	1.020	0.001
		T-test for Equality of Means	**T-test for Equality of Means**	**T-test for Equality of Means**	**T-test for Equality of Means**	
		Mean Difference	**Std.Error Difference**	**95% Confidence lower**	**95% Confidence upper**	
Accuracy	**Equal Variances assumed**	29.59	0.05025	29.3	29.811	
	Equal Variances not assumed	29.59	0.05025	28.98	30.204	

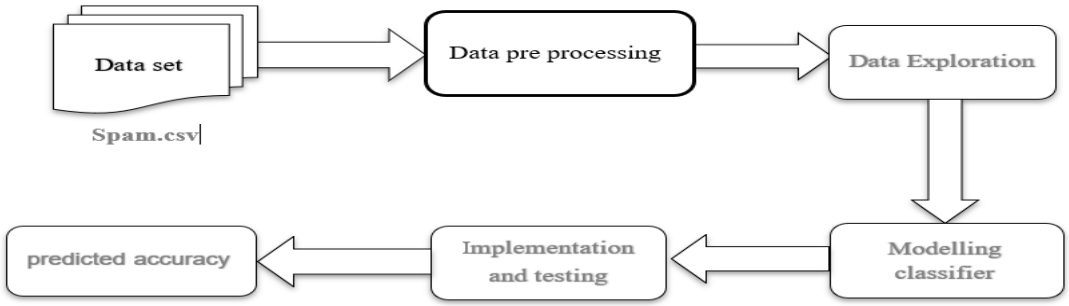

Fig 1: Machine learning classifier architecture

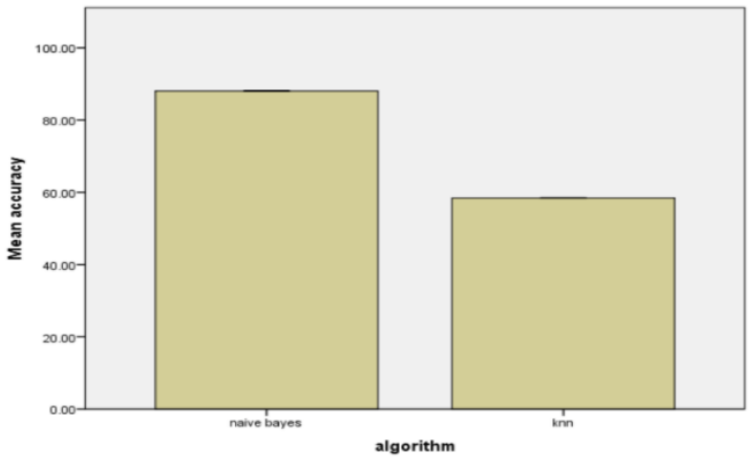

Fig 2: Simple Bar Mean of Accuracy by NBA, KNN(Error mean for naive bayes algorithm is 0.050 and for K-nearest neighbor is 0.92) Axis X: Algorithms ((Naive Bayes Algorithm, K-nearest neighbor Algorithm), Axis Y: Mean Accuracy of detection +/- 1 SD.

Table 3 shows that the accuracy of the decision tree algorithm is approximately (89%) and k-nearest neighbor algorithm is approximately (58%).The accuracy varies for different sizes in decimals

From Table 4, it is observed that mean accuracy and standard deviation for decision tree algorithms is 89.4 and 0.07071. For k-nearest neighbors the algorithm is 58.045 and 0.0707.

From Table 5, it is observed the comparison of accuracy for spam detection classification using naive bayes algorithm and k- nearest neighbor algorithm.

In performing statistical analysis, the decision tree algorithm obtained 0.07 standard deviation with 0.050 standard error while k-nearest neighbors obtained 0.070 standard deviation with 0.0500 standard error (Table 4). The significance value smaller than 0.001 showed that our hypothesis holds good. With respect to changes in the input values attributes in independent variables are profile, source, proofs and the corresponding output for dependent variables are internal user, external user.

The bar chart in Fig 3 shows the comparison of mean accuracy for decision tree algorithm and k-nearest neighbor algorithm. Naive bayes appear to produce more consistent results with minimal standard deviation.

Table 3: Accuracy performance measure (Decision tree Algorithm, K-nearest neighbor Algorithm)

Test Size	0.1	0.2
Decision Tree Algorithm	89.4	89.05
K-nearest neighbor Algorithm	58.04	58.05

Table 4: Group Statistics (Mean of Decision tree algorithm is 89.4% is more than K-nearest neighbor algorithm 58.45% and Error mean for decision tree algorithm is 0.050 and for K-nearest neighbor is 0.05000)

	DTA,KNN	**N**	**Mean**	**Std. Deviation**	**Std. Mean Error**
Accuracy	DTA	2	89.4	0.07071	0.0500
	KNN	2	58.45	0.0707	0.05000

Table 5: Independent Samples Test (decision tree algorithm appears to perform significantly better than k-nearest neighbors algorithm)

		Levene's Test for Equality of Variances	**Levene's Test for Equality of Variances**	**T-test for Equality of Means**	**T-test for Equality of Means**	**T-test for Equality of Means**
		F	**Sig.**	**t**	**df**	**Sig. (2-tailed)**
Accuracy	**Equal Variances assumed**	2.346	0.028	438.406	2	0.000
	Equal Variances not assumed	-	-	438.406	1.080	0.001
		T-test for Equality of	**T-test for Equality of**	**T-test for Equality of**	**T-test for Equality of Means**	

		Means	Means	Means	
		Mean Difference	**Std.Error Difference**	**95% Confidence lower**	**95% Confidence upper**
Accuracy	**Equal Variances assumed**	31.000	0.07071	30.69576	31.30424
	Equal Variances not assumed	31.000	0.0707	29.89686	31.60826

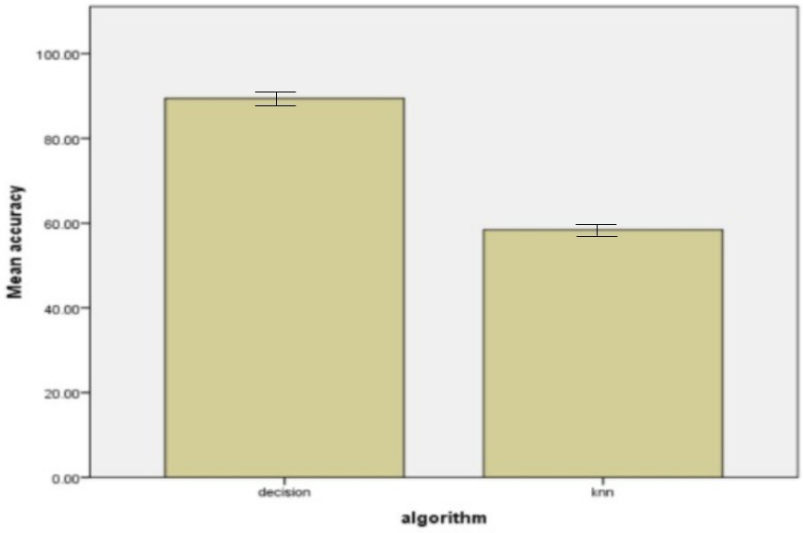

Fig 3: Simple Bar Mean of Accuracy by DTA, KNN(Error mean for decision tree algorithm is 0.050 and for K-nearest neighbor is 0.0500) Axis X: Algorithms ((Decision TreeAlgorithm, K-nearest neighbor Algorithm), Axis Y: Mean Accuracy of detection +/- 1 SD.

Table 6 shows to observe the accuracy of the random forest algorithm is approximately (89%) and k-nearest neighbor algorithm is approximately (58%).The accuracy varies for different sizes in decimals

From Table 7, it is observed that mean accuracy and standard deviation for random forest algorithm is 89.75 and 0.07071. For k-nearest neighbors the algorithm is 58.45 and 0.0707.

From Table 8, it is observed the comparison of accuracy for spam detection classification using random forest algorithm and k- nearest neighbor algorithm.

In performing statistical analysis, the random forest algorithm obtained 0.0701 standard deviation with 0.050 standard error while k-nearest neighbors obtained 0.070 standard deviation with 0.0500 standard error (Table 7). The significance value smaller than 0.001 showed that our hypothesis holds good. With respect to changes in the input values attributes in independent variables are profile, source, proofs and the corresponding output for dependent variables are internal user, external user.

Figure 4 shows the bar chart representing the comparison of mean accuracy of random forest algorithm and k-nearest neighbor algorithm. Random forests appear to produce more consistent results with minimal standard deviation.

Table 6: Accuracy performance measure (Naive Bayes Algorithm, K-Nearest Neighbor Algorithm)

Test Size	**0.1**	**0.2**
Random forest Algorithm	89.75	89.85
K-nearest neighbor Algorithm	58..4	58.5

Table 7: Group Statistics (Mean of Random Forest algorithm is 89.75% is more than K-Nearest Neighbor algorithm 58.45% and Error mean for naive bayes algorithm is 0.050 and for K-Nearest Neighbor is 0.92)

	RfA,KNN	**N**	**Mean**	**Std. Deviation**	**Std. Mean Error**
Accuracy	RFA	2	89.75	0.07071	0.5000
	KNN	2	58.45	0.07071	0.0500

Table 8: Independent Samples Test (naive bayes algorithm appears to perform significantly better than k-nearest neighbors algorithm) and naive bayes prediction models with value p<0.05. Both the algorithms are significance levels less than 0.05.

		Levene's Test for Equality of Variances	Levene's Test for Equality of Variances	T-test for Equality of Means	T-test for Equality of Means	T-test for Equality of Means
		F	Sig.	t	df	Sig. (2-tailed)
Accuracy	Equal Variances assumed	2.584	0.0002	468.953	2	0.000
	Equal Variances not assumed	-	-	468.953	2.001	0.001
		T-test for Equality of Means	T-test for Equality of Means	T-test for Equality of Means	T-test for Equality of Means	
		Mean Difference	Std.Error Difference	95% Confidence lower	95% Confidence upper	
Accuracy	Equal Variances assumed	31.3000	0.07071	30.9958	31.504	
	Equal Variances not assumed	31.3010	0.07071	29.985	31.604	

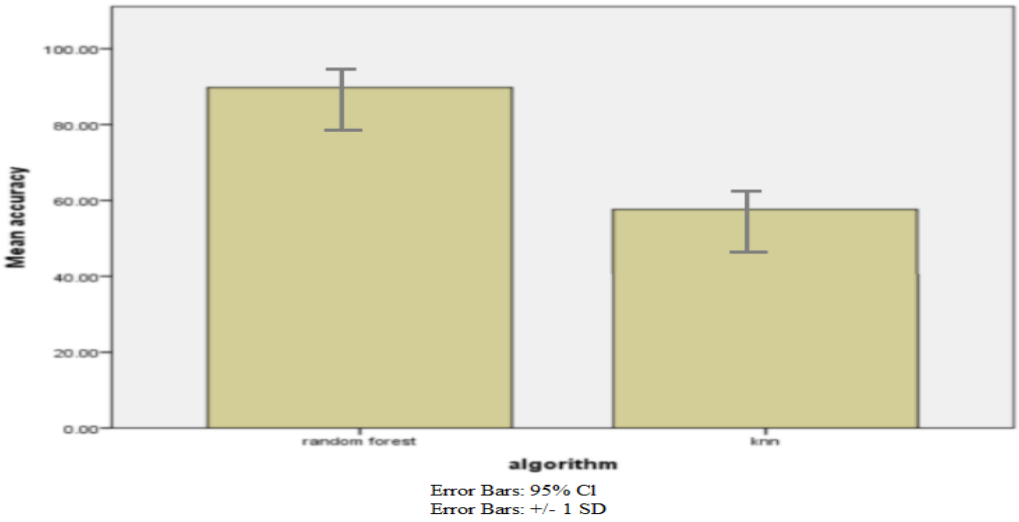

Fig. 4: Simple Bar Mean of Accuracy by RFA, KNN(Error mean for Random Forest algorithm is 0.50 and for K-nearest neighbor is 0.92) Axis X: Algorithms ((Random Forest Algorithm, K-nearest neighbor Algorithm), Axis Y: Mean Accuracy of detection +/- 1 SD

Table 9 shows to observe the accuracy of the random forest algorithm is approximately (89%) and naive bayes algorithm is approximately (88%).The accuracy varies for different sizes in decimals. From Table 10, it is observed that mean accuracy and standard deviation for naive bayes algorithm is 89.75 and 0.07071. For k-nearest neighbors the algorithm is 88.05 and 0.0707. From Table 11, it is observed the comparison of accuracy for spam detection classification using naive bayes algorithm and k- nearest neighbor algorithm.

In performing statistical analysis, the random forest algorithm obtained 0.007 standard deviation with 0.050 standard error while naive bayes obtained 0.070 standard deviation with 0.0500 standard error (Table 10). The significance value smaller than 0.001 showed that our hypothesis holds good. With respect to changes in the input values attributes in independent variables are profile, source, proofs and the corresponding output for dependent variables are internal user, external user.

Figure 6 shows the bar chart representing the comparison of mean accuracy of random forest algorithm and naive bayes algorithm. Random Forest appears to produce more consistent results with minimal standard deviation.

Table 9: Accuracy performance measure (Random Forest,Naive Bayes Algorithm)

Test Size	0.1	0.2
Random Forest Algorithm	89.75	89.85
Naive Bayes Algorithm	88.05	88.06

Table 10: Group Statistics (Mean of random forest algorithm is 89.75% is more than naive bayes algorithm 88.05% and Error mean for naive bayes algorithm is 0.050 and for K-Nearest Neighbor is 0.92)

	NBA,KNN	N	Mean	Std. Deviation	Std. Mean Error
Accuracy	RFA	2	89.75	0.07071	0.0500
	NBA	2	88.05	0.0707	0.00500

Table 11: Independent Samples Test (naive bayes algorithm appears to perform significantly better than k-nearest neighbors algorithm) naive bayes prediction models with value p<0.05. Both the algorithms are significance levels less than 0.05.

		Levene's Test for Equality of Variances	Levene's Test for Equality of Variances	T-test for Equality of Means	T-test for Equality of Means	T-test for Equality of Means
		F	Sig.	t	df	Sig. (2-tailed)
Accuracy	Equal Variances assumed	3.246	0.001	33.732	2	0.001
	Equal Variances not assumed	-	-	33.731	1.020	0.018
		T-test for Equality of Means	T-test for Equality of Means	T-test for Equality of Means	T-test for Equality of Means	
		Mean Difference	Std.Error Difference	95% Confidence lower	95% Confidence upper	
Accuracy	Equal Variances assumed	1.69500	0.05025	1.489	1.91121	
	Equal Variances not assumed	1.69500	0.05025	1.08536	2.30464	

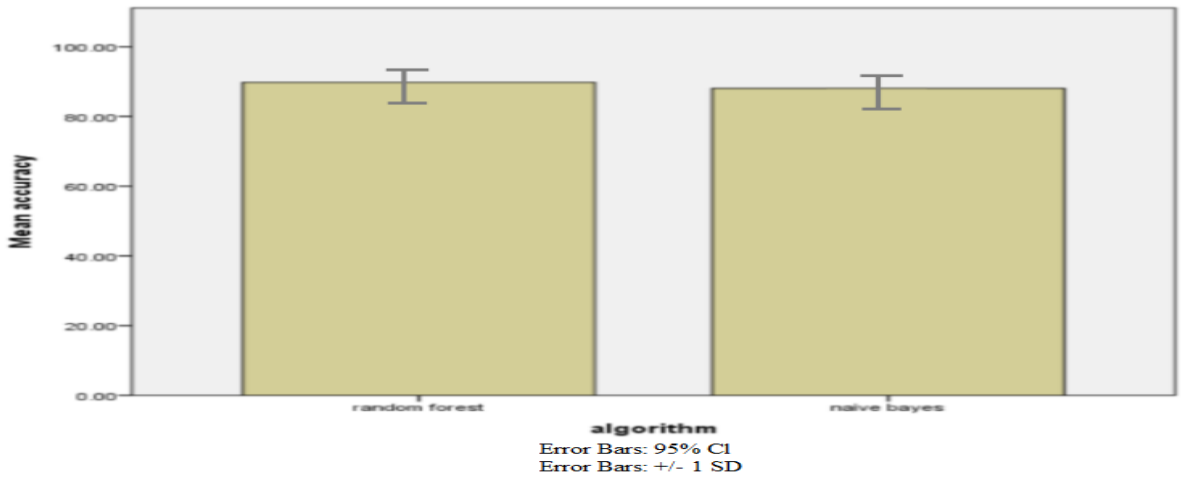

Fig. 6: Simple Bar Mean of Accuracy by RFA, NBA(Error mean for random forest algorithm is 0.050 and for naive bayes is 0.92) Axis X: Algorithms ((Random Forest Algorithm vs Naive Bayes Algorithm), Axis Y: Mean Accuracy of detection +/- 1 SD.

Table 12: Accuracy performance measure

S.NO	ALGORITHM	ACCURACY
1	Decision tree	89.4%
2	K-Nearest Neighbors	58.45%
3	Naive Bayes	88.05%
4	Random Forest	89.75%

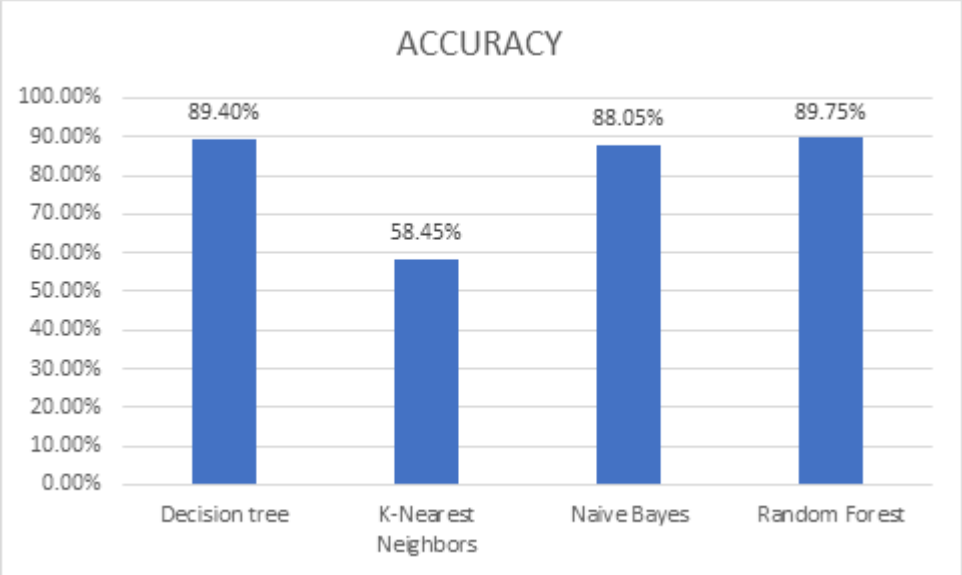

Fig. 7: Comparison of different algorithms

The proposed analysis of SMS spam detection accuracy of the Naive Bayes algorithm is 88.04% and K-Nearest Neighbors algorithm is 58.04%. Naive Bayes algorithms have better accuracy than K-Nearest Neighbors algorithms. The analysis of SMS spam detection accuracy of the Decision tree algorithm is 89.4% and K-Nearest Neighbors algorithm is 58.04%. Decision Tree algorithms have better accuracy than K-Nearest Neighbors algorithms. The analysis of SMS spam detection accuracy of the Random Forest algorithm is 89.5% and K-Nearest Neighbors algorithm is 58.45%.Random Forest algorithms have better accuracy than K-Nearest Neighbors algorithms. The proposed analysis of SMS spam detection accuracy of the Random Forest algorithm is 89.75% and Naive Bayes algorithm is 88.05%.Random Forest algorithms have better accuracy than all other algorithms is shown Figure 7 and table 12.

DISCUSSION

In this study of spam detection classification, the naïve bayes algorithm has higher accuracy (88.04%) in comparison to K-Nearest Neighbor algorithm (58.04%). There is a statistical significant difference in accuracy for two algorithms is $p<0.05$, by performing independent samples tests in the SPSS statistical tool. Naive bayes algorithm appears to produce the most consistent results with minimal standard deviation. Mean and standard deviation are also calculated using the SPSS statistical tool. Standard error difference defines the error level and it gets while performing the experiment is less than 1 when it assumes with or without variances (i.e.,0.05025) which leads to very few errors occurring during the experiment. It considered the confidence level more than 85% and resulted in getting 88%. It can observe only slight differences when assumed with and without variances.The comparison results shows that the naive bayes algorithm has appeared to be better performance than k-nearest neighbor algorithm.

From previous findings, Decision tree algorithms have better significance ($p<0.05$) when using the SPSS tool for statistical calculations. The mean accuracy and standard deviation for the naive bayes algorithm is 88.044 and 0.00707. For k-nearest neighbors, the algorithm is 58.04 and 0.0707. In

paper(Pandya 2019), Random forest and SVM(Support Vector Machine) innovative machine learning algorithms are implemented to identify the spam words on messages which gives an accuracy of 85%. In paper(Wei 2018), deep learning and naive Bayes-like innovative algorithms are implemented to identify the spam words with a ccuracy of 81% and 86%. From these two papers, it can be observed that the proposed naive bayes algorithm has appeared to be better accuracy. On the basis of literature survey it is proved that the naive bayes algorithm has better accuracy compared with K-Nearest Neighbor algorithm. There are no opposing findings for comparing naive bayes algorithms with k-nearest neighbor algorithms.

There are some limitations to this project which need to be overcome in the coming future implementation. The attributes that the dataset contains are very few to predict accuracy for spam detection classification. If it increases the independent variables(attributes), then dependent variables performance will be improved adequately. In future, if the dataset contains many attributes the classifier can work efficiently and can improve the prediction accuracy. Attributes like profile, source, proofs can result in improved accuracy and exact precision values.

CONCLUSION

The proposed analysis of SMS spam detection accuracy of the Random Forest algorithms has better accuracy than all other algorithms.

REFERENCES

1. Abbashi, M. M., Udmurt State University, A. P. Beltyukov, H. Lal, A. Q. Abbasi, Udmurt State University, AJ&K University Muzaffarabad, and Institute of Technology Karlsruhe; Women's University AJ & K; Women's University AJ & K. 2020. "SPAM DETECTION IN SHORT TEXT MESSAGES (SMS) USING WORD EMBEDDING AND TERM FREQUENCY- INVERSE DOCUMENT FREQUENCY (TF-IDF)." *XXI Century: Resumes of the Past and Challenges of the Present plus*. https://doi.org/10.46548/21vek-2020-0950-0026.

2. Anbu, R. Tamil, V. Suresh, Revathy Gounder, and Abinaya Kannan. 2019. "Comparison of the Efficacy of Three Different Bone Regeneration Materials: An Animal Study." *European Journal of Dentistry* 13 (1): 22–28.

3. Arifin, Dea Delvia, Shaufiah, and Moch Arif Bijaksana. 2016. "Enhancing Spam Detection on Mobile Phone Short Message Service (SMS) Performance Using FP-Growth and Naive Bayes Classifier." *2016 IEEE Asia Pacific Conference on Wireless and Mobile (APWiMob)*. https://doi.org/10.1109/apwimob.2016.7811442.

4. Aydogan, Murat, and Ali Karci. 2018. "Spam Mail Detection Using Naive Bayes Method with Apache Spark." *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*. https://doi.org/10.1109/idap.2018.8620737.

5. Chandrasekar, Raghavan, Shyamala Chandrasekhar, K. K. Shantha Sundari, and Poornima Ravi. 2020. "Development and Validation of a Formula for Objective Assessment of Cervical Vertebral Bone Age." *Progress in Orthodontics* 21 (1): 38.

6. "Cleaning Text Data." 2018. *Statistical Data Cleaning with Applications in R*. https://doi.org/10.1002/9781118897126.ch5.

7. Ezhilarasan, Devaraj, Velluru S. Apoorva, and Nandhigam Ashok Vardhan. 2019. "Syzygium Cumini Extract Induced Reactive Oxygen Species-Mediated Apoptosis in Human Oral Squamous Carcinoma Cells." *Journal of Oral Pathology & Medicine: Official Publication of the International Association of Oral Pathologists and the American Academy of Oral Pathology* 48 (2): 115–21.

8. Felicita, A. Sumathi. 2017a. "Quantification of Intrusive/retraction Force and Moment Generated during En-Masse Retraction of Maxillary Anterior Teeth Using Mini-Implants: A Conceptual Approach." *Dental Press Journal of Orthodontics* 22 (5): 47–55.

9. "Fluoride, Fluoridated Toothpaste Efficacy and Its Safety in Children - Review." 2018. *International Journal of Pharmaceutical Research* 10 (04). https://doi.org/10.31838/ijpr/2018.10.04.017.

10. Gupta, Mehul, Aditya Bakliwal, Shubhangi Agarwal, and Pulkit Mehndiratta. 2018. "A Comparative Study of Spam SMS Detection Using Machine Learning Classifiers." *2018 Eleventh International Conference on Contemporary Computing (IC3)*. https://doi.org/10.1109/ic3.2018.8530469.

11. Jain, Ashish R. 2017. "Clinical and Functional Outcomes of Implant Prostheses in Fibula Free Flaps." *World Journal of Dentistry* 8 (3): 171–76.

12. Kane, Sean P., Phar, and BCPS. n.d. "Clinical Tools and Calculators for Medical Professionals - ClinCalc." Accessed March 29, 2021. https://clincalc.com/.

13. Khanday, Shahbaz Ahmad. 2019. "A Survey on Various Machine Learning and Deep Learning Algorithms Used for Classification of Spam and Non-Spam Emails." *International Journal for Research in Applied Science and Engineering Technology*. https://doi.org/10.22214/ijraset.2019.4568.

14. Kołcz, Aleksander. 2017. "Text Mining for Spam Filtering." *Encyclopedia of Machine Learning and Data Mining*. https://doi.org/10.1007/978-1-4899-7687-1_828.

15. Kumar, Santhosh. 2017a. "Relationship between Dental Anxiety and Pain Experience during Dental Extractions." *Asian Journal of Pharmaceutical and Clinical Research* 10 (3): 458.

16. Mathew, Mebin George, S. R. Samuel, Ashu Jagdish Soni, and Korishettar Basavaraj Roopa. 2020. "Evaluation of Adhesion of Streptococcus Mutans, Plaque Accumulation on Zirconia and Stainless Steel Crowns, and Surrounding Gingival Inflammation in Primary Molars: Randomized Controlled Trial." *Clinical Oral Investigations*, 1–6.

17. Mehta, Meenu, Deeksha, Devesh Tewari, Gaurav Gupta, Rajendra Awasthi, Harjeet Singh, Parijat Pandey, et al. 2019. "Oligonucleotide Therapy: An Emerging Focus Area for Drug Delivery in Chronic Inflammatory Respiratory Diseases." *Chemico-Biological Interactions* 308 (August): 206–15.

18. Nair, Manisha, Ganesh Jeevanandan, Vignesh R, and Subramanian Emg. 2018. "Comparative Evaluation of Post-Operative Pain after Pulpectomy with K-Files, Kedo-S Files and Mtwo Files in Deciduous Molars -a Randomized Clinical Trial." *Brazilian Dental Science* 21 (4): 411.

19. Pandya, Darshit. 2019. "Spam Detection Using Clustering-Based SVM." *Proceedings of the 2019 2nd International Conference on Machine Learning and Machine Intelligence*. https://doi.org/10.1145/3366750.3366754.

20. Patterson, Anthony. n.d. "Digital Youth, Mobile Phones and Text Messaging." *The Routledge Companion to Digital Consumption*. https://doi.org/10.4324/9780203105306.ch8.

21. Pc, J., T. Marimuthu, and P. Devadoss. 2018. "Prevalence and Measurement of Anterior Loop of the Mandibular Canal Using CBCT: A Cross Sectional Study." *Clinical Implant Dentistry and Related Research*. https://europepmc.org/article/med/29624863.

22. Ponnulakshmi, R., B. Shyamaladevi, P. Vijayalakshmi, and J. Selvaraj. 2019. "In Silico and in Vivo Analysis to Identify the Antidiabetic Activity of Beta Sitosterol in Adipose Tissue of High Fat Diet and Sucrose Induced Type-2 Diabetic Experimental Rats." *Toxicology Mechanisms and Methods* 29 (4): 276–90.

23. Ramadurai, Neeraja, Deepa Gurunathan, A. Victor Samuel, Emg Subramanian, and Steven J. L. Rodrigues. 2019. "Effectiveness of 2% Articaine as an Anesthetic Agent in Children: Randomized Controlled Trial." *Clinical Oral Investigations* 23 (9): 3543–50.

24. Ramesh, Asha, Sheeja Varghese, Nadathur D. Jayakumar, and Sankari Malaiappan. 2018. "Comparative Estimation of Sulfiredoxin Levels between Chronic Periodontitis and Healthy Patients - A Case-Control Study." *Journal of Periodontology* 89 (10): 1241–48.

25. R, Hannah, R. Hannah, Pratibha Ramani, Arvind Ramanathan, Jancy Merlin R, S. Gheena, Abilasha Ramasubramanian, and K. Monika. 2020. "CYP2 C9 Polymorphism among Patients with Oral Squamous Cell Carcinoma and Its Role in Altering the Metabolism of Benzo[a]pyrene." *Oral Surgery, Oral Medicine, Oral Pathology and Oral Radiology*. https://doi.org/10.1016/j.oooo.2020.06.021.

26. Samuel, Melvin S., Jayanta Bhattacharya, Sankalp Raj, Needhidasan Santhanam, Hemant Singh, and N. D. Pradeep Singh. 2019. "Efficient Removal of Chromium(VI) from Aqueous Solution Using Chitosan Grafted Graphene Oxide (CS-GO) Nanocomposite." *International Journal of Biological Macromolecules* 121 (January): 285–92.

27. Samuel, Srinivasan Raj. 2021. "Can 5-Year-Olds Sensibly Self-Report the Impact of Developmental Enamel Defects on Their Quality of Life?" *International Journal of Paediatric Dentistry / the British Paedodontic Society [and] the International Association of Dentistry for Children* 31 (2): 285–86.

28. Sridharan, Gokul, Pratibha Ramani, and Sangeeta Patankar. 2017. "Serum Metabolomics in Oral Leukoplakia and Oral Squamous Cell Carcinoma." *Journal of Cancer Research and Therapeutics* 13 (3): 556–61.

29. Sridharan, Gokul, Pratibha Ramani, Sangeeta Patankar, and Rajagopalan Vijayaraghavan. 2019. "Evaluation of Salivary Metabolomics in Oral Leukoplakia and Oral Squamous Cell Carcinoma." *Journal of Oral Pathology & Medicine: Official Publication of the International Association of Oral Pathologists and the American Academy of Oral Pathology* 48 (4): 299–306.

30. Trivedi, Shrawan Kumar. 2016. "A Study of Machine Learning Classifiers for Spam Detection." *2016 4th International Symposium on Computational and Business Intelligence (ISCBI)*. https://doi.org/10.1109/iscbi.2016.7743279.

31. Vijayashree Priyadharsini, Jayaseelan. 2019. "In Silico Validation of the Non-Antibiotic Drugs Acetaminophen and Ibuprofen as Antibacterial Agents against Red Complex Pathogens." *Journal of Periodontology* 90 (12): 1441–48.

32. Vijayashree Priyadharsini, J., A. S. Smiline Girija, and A. Paramasivam. 2018. "In Silico Analysis of Virulence Genes in an Emerging Dental Pathogen A. Baumannii and Related Species." *Archives of Oral Biology* 94 (October): 93–98.

33. "Website." n.d. Accessed March 29, 2021. https://www.kaggle.com/del=816d012218d4636b/sms-ham-spam-detection.

34. Wei, Qijia. 2018. "Understanding of the Naive Bayes Classifier in Spam Filtering." https://doi.org/10.1063/1.5038979.