A Comparative Analysis of SMS Spam Detection Employing Machine Learning Methods

Humaira Yasmin Aliza
Department of computer science & Engineering
Daffodil International University
Dhaka, Bangladesh
humaira15-2460@diu.edu.bd

Kazi Aahala Nagary
Department of computer science & Engineering
Daffodil International University
Dhaka, Bangladesh
aahala15-2410@diu.edu.bd

Eshtiak Ahmed
Faculty of Information Technology
and Communication Sciences
Tampere University, Tampere, Finland
eshtiak.ahmed@tuni.fi

Kazi Mumtahina Puspita
Department of computer science & Engineering
Daffodil International University
Dhaka, Bangladesh
mumtahina15-11407@diu.edu.bd

Khadiza Akter Rimi
Department of computer science & Engineering
Daffodil International University
Dhaka, Bangladesh
khadiza15-2411@diu.edu.bd

Ankit Khater*
Department of Computer Science & Engineering
Jadavpur University
Jadavpur, India
ankitkhaterak@gmail.com

Fahad Faisal
Department of Computer Science & Engineering
Daffodil International University
Dhaka, Bangladesh
fahad.cse@diu.edu.bd

Abstract— **In recent times, the increment of mobile phone using has resulted in a huge number of spam messages. Spammers continuously apply more and more new tricks that cause managing or preventing spam messages a challenging task. The aim of this study is to detect spam message to prevent different cybercrimes as spam messages has become a security threat nowadays. In this paper, we contributed to previous studies on SMS spam problem to perform a better accuracy using several different techniques such as Support Vector Machine, K-Nearest Neighbor, Naïve Bayes, Random Forest, Logistic Regression and some more. Our result indicated that Support Vector Machine achieved the highest accuracy of 99%, so that it might be useful as an effective machine learning system for future research.**

Keywords— Spam, Ham, Classifier, Accuracy, Sensitivity, Specificity.

I. Introduction

The term 'SMS SPAM' refers to unsolicited and objectionable messages sent via SMS [1]. In the beginning of SMS spams generally contained simply commercial advertisements, on the contrary current spams usually contain several malwares and spywares embedded in attachments or website links with which a receiver often becomes an unknown victim of numerous cybercrimes [2]. At present, spam has become a major problem on Facebook, WhatsApp and other social messaging applications [3]. Still people fall for spam messages and their personal information, account number, important and confidential documents, passwords get revealed [4]. Though the network of these social platforms can identify spam messages and prevent them, scenario is different in email where this is still a serious issue. Hence, those issues brought on by spam could extend from basic annoyances to critical security issues [5, 31]. The current studies show that mobile SMS spam filtering procedures have stayed at their underlying phase of classification, for instance unequivocal number obstructing or the character string similarity [6]. Dreaded spam emails flood inboxes which require lots of time to detect and delete. Moreover, it causes problems such as weak service performance, increase cost; company operative throughput etc. to deal with the spams, sometimes company hires more employees that caused an increase in company budget [7]. Spamming has been a relevant problem in Far East countries since year 2001 [8]. By 2005, an amount more than 66% of all SMS sent over the Internet were spams, which increased to 70% by 2010. However by 2015 it became 73% which is considerable whereas today it basically expanded alarmingly to 85% of all SMS. SMS spam is considered as a bigger social media problem [9].

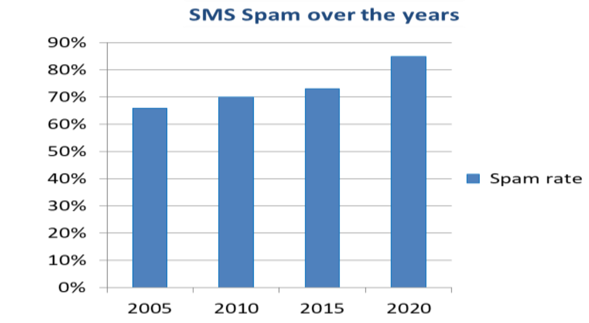

Fig. 1. Statistical Representation of SMS Spam

The public authority of a few nations has fostered the Information Security Manual, in view of International Standards ISO/IEC 27001:2013 and ISO/IEC 27002:2013, which contains some separated arrangements on spam [10]. Conversely, these laws have helped to reduce but not quit SMS spam problem. Likewise in view of the accessibility of cell phone, spam-filtering programming is constrained [11]. So experts consider that SMS spam can only get controlled through the combination of technical and legal measures. In this regard a fully automated system to recognize spam messages and filtering them out could be a great solution of these security issues [12]. Therefore, this research proposed a study using a dataset consisting of spam message and legitimate messages to distinguish spam from legitimate texts.

II. Literature Review

In this section, comparable investigations that were distributed by different researchers have been analyzed to call attention to the issues which actually must be addressed and to feature the distinctions with examination. A. Karim et al.

(2021) [1] outlines a comprehensive systematic review of Artificial Intelligence (AI) and Machine Learning (ML) approaches for intelligent spam email detection, and provides an intelligent method based on the quantity of relevant spam messages. In his article, he looked at four aspects of the email structure that could be utilized for intelligent analysis, classified all procedure, analyzed, summarized, and pointed out that some methods, such as SVM and Nave Bayes, are in great abundance. He further led to the realization that single-algorithm anti-spam methods are relatively widespread, hence studies towards hybrid and multi-algorithm methods has a huge amount of potential. Suparna Das Gupta et al. (2021) [9] mentioned in their study that they have developed system which will identify malicious messages and if the message Spam or Ham. The authors created a dictionary using the Turn Frequency Inverse Document Frequency (TF-IDF) Vector algorithm in which they obtained 95% of accuracy. L. G. a. Jun et al. (2020) [10] used K-NN classification, DT and Logistic Regression in his research. His results prove that LR classification performance achieving maximum accuracy of 99%. H. Yang et al. (2019) [11] got 98.48% accuracy by using a model named multi-modal architecture based on model fusion (MMA-MF), Convolutional Neural Network (CNN) model and Long Short-Term Memory (LSTM) model in their work. O. Abayomi-Alli et al. (2019) [13] published a review paper in which he conducted approximately AI methods for example Naïve Bayes, SVM, Random Forest, Dendritic Cell Algorithm, AIS, etc. and have shown optimal performance result with upper accuracy. M. Bassiouni et al. (2018) [14] has reported that their finest performance was realized using Random Forest technique, achieving 95.45% accuracy. A survey was conducted by S. Jeong et al. (2016) [15] TSP-Filtering for Random Forest offers a powerful spam-classification performance of 92.1%, according to the results. E. Ezpeleta et al. (2016) [16] shown in his work that terms of accuracy Nearly 98.76% of the time, the SMO approach was the best and using polarity a 98.91% of accuracy is obtained. Vivekanandam B. et al. (2021) [24] explains about using ml algorithms to tackle functional challenges by preserving the selection and evolution technicians in computational modeling. The presented scheme is flexible and adaptive throughout workforce estimations in the training phase, so it provides the maximum suited probability of resolving feature extraction complexities during the training. Goswami et al. (2019) [26] measures the effectiveness of several supervised machine learning techniques for eliminating Ham and Spam messages, including the naive Bayes Technique, support vector machines approach, and maximum entropy method. Generating a spam filter combining discrete and continuous probability distributions is just the strategy proposed in this study. When compared to SVM, The Naive Bayes fared remarkably well in his work. Dubey et al. (2021) [27] provided a machine learning approach for SMS Spam filtering in his article, which was based on computations such as Nave Bayes and Support vector machines, and also precision and accuracy, that was calculated through using perplexity framework. He observed that perhaps the Naive Bayes performed exceptionally really well compared to SVM in screening the Ham and Spam messages after evaluating respective implementation.

III. RESEARCH METHODOLOGY

A systematic overview has been added to provide a better idea about this proposed work.

A. Data Collection

Our utilized data in this study was collected from UCI repository [17] that contains 4,827 legitimate messages and 747 mobile spam messages, a total of 5,574 short messages. To the best of our knowledge, it is the largest available SMS spam corpus that currently exists. The following table shows the basic description of the collected data.

TABLE I. A DESCRIPTION OF TAKEN DATASET

Message	Amount	Datatype
Ham	4,827	Float64
Spam	747	Float64
Total	5,574	Float64

In this dataset, there's almost 4900 of the samples are ham and 747 samples are spam and the format of data is in floating-point occupying 64 bits of computer memory. Figure 1 shown the statistic and the ratio of total class and messages have shown in Fig. 2.

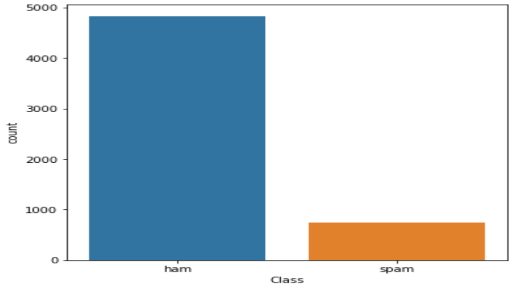

Fig. 2. The ratio of HAM and SPAM values

B. Overview of the Proposed Model

The main objective of this research is to provide strong baseline results on spam detection and compare the results with some existing literatures. For that, we have introduced several well-known machine learning algorithms to perform spam filtering with the dataset SMS Spam Collection. In this phase, after collecting the dataset data preprocessing is performed. Because the information was given ambiguities, inaccuracies, and overabundance which must be cleansed upfront, data pre-processing is by far the most crucial phase in research frameworks.

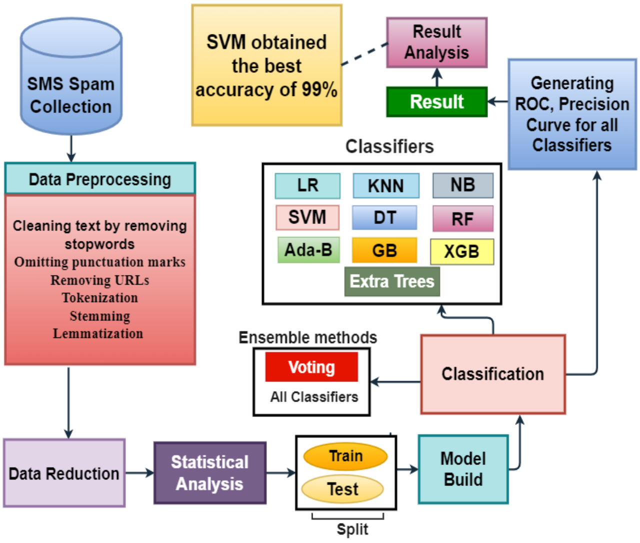

Fig. 3. A flow chart of explained approach

The text messages are converted into a predictable and analyzable format for training the models in data preprocessing, and the cleaning procedures are carried out by eliminating stop words from texts, omitting punctuation marks such as., ! $ () * % @, removing URLs, tokenization, stemming, and lemmatization. On the basis of the two classes provided, we run some statistical analysis. Some words in the dataset had mixed-case occurrences and there was insufficient evidence for the artificial machine to understand the parameters for the less common variant adequately. As a result, lowercasing was an essential technique to tackle backup and recovery complications. The characteristics defined for categorizing preparation and framework execute include extraction, following which these classification framework categorize the material as spam or ham. Next, the preprocessed dataset is divided into two parts: training and testing. As mentioned, experimented with several ML classifiers, the models are built in the next step. For implementing the classification task, two ensemble methods are performed. Finally, different performance matrices are evaluated to analysis the result and compare the classifiers (see Fig. 3) and Pseudocode 1 [12].

Pseudocode 1: Pseudocode of our proposed work

```
    Step 1: Function Pre-processing ()
    Step 2: import dataset
    Step 3: Removing stopwards
    Step 4: Omitting punctuation marks
    Step 5: End Pre-processing ()
    Step 6: Function TrainTestSplit ()
    Step 7: Go to Step 5
    Step 8: End TrainTestSplit ()
    Step 9: Function BestClassifiersSelectionApproach ()
    Step 10: Choose 9 algorithms (LR, KNN, SVM, NB, DT, AB, RF, GB and XGB)
    Step 11: For i = 0: 9
    Step 12: Predict class of data
    Step 13: Evaluate result
    Step 14: Go to Step 11
    Step 15: Comparison among overall outcomes
    Step 16: Recommend the best model
    Step 17: End BestClassifiersSelectionApproach ()
```

IV. MACHINE LEARNING APPROACHES

In this section, the classifiers that are used in this experiment have been briefly discussed. Nine classifier namely logistic regression (LR) [1] [10] [13] [26], K-Nearest Neighbor (KNN) [10], Support Vector Machine (SVM) [1] [13] [26] [27], Naïve Bayes (NB) [1] [13] [26] [27], Decision Tree (DT) [1] [10], Ada-Boost (AB), Random Forrest (RF) [1] [13] [14] [15], Gradient Boost (GB), Extreme Gradient Boost (XGB) are introduced and compared their performance in term of accuracy [30] and other classification matrices [32, 33].

A. Logistic Regression

Logistic regression [18] is a classification algorithm that is used when the value of the target variable is categorical in nature. Logistic regression is most commonly used when the data in question has binary output, so when it belongs to one class or another, or is either a 0 or 1. It predicts the variable value of multi classification such as y ∈ [0, 1, 2, 3] [19].

$$\log\ (p\ (X)\ 1-p\ (X)) = \beta_0+\beta_1 X \qquad (1)$$

$$\log\left(\frac{p(X)}{1-p(x)}\right) = \beta_0|\beta_1 X \qquad (2)$$

$$Y = \beta_0|\beta_1 X \qquad (3)$$

Here, Y = numerical value, X = given value and p = probability. The coefficients β_0 and β_1 must be estimated based on the available training data.

B. K Nearest Neighbors

The K-nearest neighbors (KNN) [19] is a simple, easy-to-implement supervised machine learning algorithm that can be used to solve both classification and regression problems. The KNN algorithm assumes that similar things exist in close proximity or in Other words, similar things are near to each other. KNN predicts its label using the same as the class and its input as a new input training set. (X, Y) training observation and learning the function h: X ->Y, so that an observation x, h(x) can set y value. Suppose A1 (x1, y1) and point p2 (x2, y2). Euclidean distance for these two is given in Equation 4 [25].

$$\mathrm{E(D)} = \sqrt{}((x1-x2)^2+(y1-y2)^2) \qquad (4)$$

Here, (x1, x2) is the feature vector of point x and (y1, y2) is of point y. E (D) denotes the Euclidean distance.

C. Support Vector Machines (SVM)

Support Vector Machines are supervised learning models [21] that evaluate data for classification and regression analysis. It is one of the most reliable linear classification prediction methods. By applying the kernel method, SVMs may conduct non-linear classification effectively, effectively translating their inputs into large feature sets. A data point is represented as a p-dimensional vector in this classification, and the method's purpose is to distinguish those points using a (p - 1) dimensional Hyper-plane [22]. The hyper-plane is generated using SVM so that the distance between it and the closest piece of data per each side is maximized.

D. Naïve Bayes (NB)

Naive Bayes is a straightforward approach [23] for building classifiers: models that assign class labels to problem samples by describing the attributes as a vector of values and selecting class labels from a constrained set. The number of parameters required for Naive Bayes classifiers is linear in the variety of features in a learning task, making them extremely adaptable. Basic Bayes classifications have performed admirably in a variety of complicated real-world scenarios, despite their naive structure and extremely simplified hypotheses. The benefit of Naïve Bayes classifier is that it just takes a modest quantity of training models to calculate the classification factors. The probabilistic model could well be decomposed by Bayes' theorem as follows [23]:

$$p(C_K|\ X) = \frac{P(C_k)p(X\ |C_K)}{P(X)} \qquad (5)$$

Here, $P\ (C_K|\ X)$ represents the posterior probability for class (target) involves selecting (attribute), $P(C_k)$ represents the prior probability of class, $P(X\ |C_K)$ reflects the possibility of predictor labeling, while $P(X)$ provides the prior probability of predictor. Ultimately the above formula can be expressed as follows in simple terms, using Bayesian probability terminology:

$$\text{posterior} = \frac{\text{prior} * \text{likelihood}}{\text{evidence}} \tag{6}$$

E. Decision Tree (DT)

The Decision Tree is a continuous supervised classifier [24] that is being used to solve classification and regression challenges, however it is often utilized to overcome classification task. The algorithm in a decision tree starts at root node of the tree to estimate the class of a given set of data. The method compares the features of the original component with the values of the record (real dataset) characteristic, and then goes to the next node after the branch depending on the comparison. The algorithm continues one step farther for the next node, comparing the characteristic with those of other sub-nodes. This cycle repeats until the tree's leaf destination is reached.

$$\mathrm{H(s)} = -\mathrm{P}\log_{2(\mathrm{p}+)} - \mathrm{P}\log_{2(\mathrm{p}-)} \tag{7}$$

Where, H is highest value, P denotes probability, $\mathrm{p}+$ means % of positive class and $\mathrm{p}-$ means % of negative class. The reason for choosing this method is that it closely resembles people's capacity to think while drawing conclusions, making it simple to comprehend also because it looks like a tree structure. In Decision Tree, entropy controls splitting of data, affects the Tree in drawing its boundaries [25].

F. Random Forest (RF)

Random forest is the most commonly used algorithm due to its simplicity and diversity [26, 9]. This is a robust, convenient machine learning method which consistently generates excellent results. Random forest has the benefit of being able to solve simultaneously regression and classification problems. Random forest algorithms have the added benefit of identifying the key results of each parameter in prediction simplified. Whereas the trees grow, the random forest contributes extra variability to the model. When splitting a node, it looks for the best feature in a randomized selection of attributes rather than the most essential feature. It is a vast range of possibilities that generally leads to a superior model [23].

$$\begin{aligned}\mathrm{Gini} &= 1 - \sum_{\mathrm{i}=1}^{\mathrm{c}} (\mathrm{p_i})^2 \\ &= 1 - [(\mathrm{P}+)^2 + (\mathrm{P}-)^2\end{aligned} \tag{8}$$

In which P+ denotes the possibility of a particular class while P_ indicates the possibility of a negative class.

G. AdaBoost (AB)

AdaBoost or adaptive boosting is a repetitive combination method. The Ada-Boost classifier combines multiple poor performance classifiers to create a powerful method that gives it a strong classification of high accuracy. Though Ada-Boost is sensitive to noise data it is easy to implement [23].

$$\mathrm{D_{t+1}(i)} = \frac{\mathrm{D_t(i)}\exp\left(-\alpha_\mathrm{t} \mathrm{y_i} \mathrm{h_t}(\mathrm{X_i})\right)}{\mathrm{Z_t}} \tag{9}$$

Where, $(\mathrm{X_i}, \mathrm{y_i})$: first training sample, $\mathrm{h_t}$ is hypothesis, exp denotes Euler's e: 2.71828, α_t = weight for the classifier, $\mathrm{Z_t}$ is a normalization factor and $\mathrm{D_{t+1}}$ is a distribution. The objective is to contribute scores towards both classifiers plus relevant data (samples) in a quite manner which filters being driven to emphasize upon tricky occurrences. Each procedure is carried out in such a systematic fashion, with both the variable parameters being modified through each phase as the technique is iterated.

H. Gradient Boosting (GB)

The Gradient Boosting Classification [21] algorithm is a collection of machine learning techniques that combine many inferior teaching methods to generate a powerful prediction model. Because of its proficiency in categorizing complicated datasets, this methodology has gained popularity. Gradient Boosting Classifier uses a poor hypothesis or training method and enhances it through a sequence of modifications. Gradient boosting algorithms are preferred because they can be utilized for more than binary classification tasks; they can also be used for multi-class regression and classification problems [23].

$$\text{value} = \frac{\sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{R}}{\sum_{\mathrm{i}=1}^{\mathrm{n}} [\mathrm{p} * (1 - \mathrm{P})]} \tag{10}$$

Here, R is Residual and P is previous probability.

I. Extreme Gradient Boost (XGBoost)

XG-Boost is a gradient boosting approach [27] that employs a decision-tree-based ensemble Machine Learning algorithm. It is a decentralized framework that has been developed to be highly effective, adaptable, and accessible. The reason behind choosing this classifier is XG-Boost offers a parallel tree boosting method which solves several data science tasks in a fast and precise manner [27].

$$\text{value} = \frac{\sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{R}}{\sum_{\mathrm{i}=1}^{\mathrm{n}} [\mathrm{p} * (1 - \mathrm{P})] + \lambda} \tag{11}$$

Where, R is Residual, P is Previous probability and Lambda (λ) denotes L2 regularization term on weights. By boosting this parameter, the system will become more severe.

J. Extra Tree Classifier

The Extra Trees Classifier [27], also known as the Extremely Randomized Trees Classifier, is a machine learning ensemble technique that integrates many de-correlated decision tree outcomes collected in a forest as a result of classifying them. It is similar to a random forest classifier and differs mainly in how forest decision trees are constructed.

$$\text{Entropy} = \sum_{\mathrm{i}=1}^{\mathrm{c}} -\mathrm{p_i} * \log_2(\mathrm{p_i}) \tag{12}$$

Here, c indicates the number of unique class labels while $\mathrm{p_i}$ is the proportion of rows with output label is i.

E. Ensemble Methods

A. Voting

A voting classifier [23] is a machine learning technique that's also learned to use a combination of approaches that predicts output based on the output of the chosen class with the highest probability.

1. Soft Voting: For this experiment, we implemented soft voting [26]. The output class in soft voting is the prediction based on the average probability assigned to that class. Assume that given some input to three models, the prediction probability for class A = (0.30, 0.47, 0.53) and B = (0.30, 0.47, 0.53) and C = (0.30, 0.47, 0.53) and D = (0.30,

0.47, (0.20, 0.32, 0.40). So, with an average of 0.4333 for class A and 0.3067 for class B, class A is clearly the winner because it had the highest probability averaged by each classifier.

$$\hat{y} = \arg\max_{i} \sum_{j=1}^{m} w_j \chi_A\big(C_j(x) = i\big) \qquad (13)$$

In above equation, m = each instances, j denotes the decision profile of learner, χ_A is the feature function whereas A is the set of distinct number of classes, C_j represents the classifer, w_j represents the weight associated with the prediction of the classifier.

V. Results and discussions

A. Hyperparameter Tuning

To train the models, the dataset must be divided into two sections: training and testing. Table II represents the training parameters that have been used to build and train the applied algorithms [23] [25]. In this term, different types of parameters are used, however, LR, NB, RF, GB did not contain any parameter.

TABLE II. PARAMETERS USED

Applied Algorithms	Parameters
LR	default
KNN	n_neighbors=7
SVC	probability=True
NB	default
DT	max_depth=6, random_state=123,criterion='entropy'
RF	default
AB	base_estimator = None
GB	default
XGB	objective ='reg:linear', colsample_bytree = 0.3, learning_rate = 0.1, max_depth = 5, alpha = 10, n_estimators = 10
ET	n_estimators=100, random_state=0

B. Experimented Results of Intrioduced Algorithms

It is clear that SVM achieved the best result between all algorithms with the accuracy of 99%, snesitivity of 93% and specificity of 95%. The second best accuracy is obtained by XG Boost and Extra Trees Classifier which gives 98%, %, snesitivity of 91% and specificity of 94%. After that accuracy of 97% is resulted by five classifiers, among them is NB, RF, AB, GB and XGB. LR gives 96% and Decision Tree 95% accuracy which is not near as bad. Lastly the lowest performed accuracy obtained by KNN algorithm gives the indication that KNN is least suitable performance measures for the task whereas, SVM is the best suited perfomance measure. It recorded an accuracy of 90%, Sensitivity of 60%, Specificity of 65%, F1-Score of 95% and Precision of 90%. Rest of the classifier performed with moderate spam detection accuracy.

TABLE III. CLASSIFICATION RESULTS BETWEEN DIFFERENT CLASSIFIERS

Classifier	Accuracy (%)	Recall (%)	Specificity (%)	F1 Score (%)	Precision (%)
LR	96	85	93	98	96
KNN	90	60	65	95	90
SVC	99	91	95	99	98
NB	97	87	92	98	97
DT	95	82	87	97	95
RF	97	90	93	98	97
AB	97	90	94	99	97
GB	97	90	93	99	97
XGB	98	91	94	99	97
Extra T.	98	91	94	99	97
voting	97	88	92	98	97

C. Scores of Area Under Curve (AUC)

AUC score [29, 34] for LR, KNN, SVC, NB, DT, RF, AB, GB, XGB, ET algorithms. From Table IV, see the highest value of AUC score is 99% and lowest value is 82%. And get highest AUC Score from LR, SVC, NB, RF, AB, GB, XGB, ET and lowest AUC score from the KNN model.

TABLE IV. RESULTS OF AREA UNDER CURVE

Algorithms	AUC Score (%)
LR	99
KNN	82
SVC	99
NB	99
DT	87
RF	99
AB	99
GB	99
XGB	99
ET	99
Voting	96

D. Comparison with some existing literature

Table IV shows that the majority of the existing works generates a comparatively lower performance compared with our proposed work. The overall result was shown in Table V.

TABLE V. COMPARISONS WITH EXISTING LITERATURE

Paper	Dataset	Model	Accuracy
Karim et al. (2021) [1]	Collected several databases from IEEE, Google Scholar, Science Direct, Springer, Wiley, ACM and selected papers based on the listed index terms	CNN, NB, DT, RF, LR SVM	92%
Gupta et al. (2021) [9]	Dataset collected from Kaggle Repository	TF-IDF Vector algorithm	95.90%
Yang et al. (2019) [11]	Collected three types of email datasets from Enron corpus, Personal Image Ham, Personal Image Spam and Spam Archive Image Spam	MMA – MF, CNN, LSTM	98.48%
Bassiouni et al. (2018) [14]	Collected data from a database called spam base UCI	RF	95.45%
Our Work	Collected from UCI repository containing 4,827 legitimate messages and 747 mobile spam messages, a total of 5,574 short messages	SVC	99%

VI. CONCLUSION

SMS Spam detection has become a much needed course to secure our social media. However it turned out to be a difficult task nowadays. Inconvenience in the development of algorithms in this particular field of research is the scarcity of reliable datasets. The text is expanded by idioms and abbreviations, also the small number of features per message to be removed. To fulfill some of those requirements, we have performed statistical representations that are related to the collected dataset. Compared the performance in several well established machine learning methods and found that Support vector machine classifier accomplished the highest accuracy of 99%. This propose approach can be a great solution to assist on preventing spam messages, cyber-crime, ensuring cyber security and save money without any human involvement. For future research, we intent to extend current proposition using deep Learning [34] algorithms.

REFERENCES

[1] A. Karim, S. Azam, B. Shanmugam, and K. Kannoorpatti, "An unsupervised approach for content-based clustering of emails into spam and Ham through multiangular feature formulation," IEEE Access, vol. 9, pp. 135186–135209, 2021.

[2] S. Smadi, N. Aslam, and L. Zhang, "Detection of online phishing email using dynamic evolving neural network based on reinforcement learning," Decision Support Systems, vol. 107, pp. 88–102, 2018.

[3] W. Mao, Z. Cai, D. Towsley, Q. Feng, and X. Guan, "Security importance assessment for system objects and malware detection," *Computers & Security*, vol. 68, pp. 47–68, 2017.

[4] K. Zainal, N. Sulaiman, and M. Jali, ''An analysis of various algorithms for text spam classification and clustering using RapidMiner and Weka,'' Int. J. Comput. Sci. Inf. Secur., vol. 13, no. 3, pp. 66–74, 2015.

[5] E.-S. M. El-Alfy and A. A. AlHasan, ''Spam filtering framework for multimodal mobile communication based on dendritic cell algorithm,'' Future Generat. Comput. Syst., vol. 64, pp. 98–107, 2016.

[6] L. Chen, Z. Yan, W. Zhang, and R. Kantola, ''TruSMS: A trustworthy SMS spam control system based on trust management,'' Future Generat. Comput. Syst., vol. 49, pp. 77–93, Aug. 2015.

[7] J. M. G. Hidalgo, T. A. Almeida, and A. Yamakami, ''On the validity of a new SMS spam collection,'' in Proc. 11th Int. Conf. Mach. Learn. Appl. (ICMLA), Dec. 2012, pp. 240–245.

[8] C. F. M. Foozy, R. Ahmad, and M. F. Abdollah, ''A framework for SMS spam and phishing detection in malay language: A case study,'' Int. Rev. Comput. Softw., vol. 9, no. no. 7, pp. 1248–1254, 2014.

[9] SD Gupta, S Saha, SK Das, " SMS Spam Detection Using Machine Learning", Journal of Physics: Conference Series, Vol.1797, no. 1, p. 012017, 2021.

[10] Luo GuangJun, Shah Nazir, Habib Ullah Khan, Amin Ul Haq, "Spam Detection Approach for Secure Mobile Message Communication Using Machine Learning Algorithms", Security and Communication Networks, vol. 2020, Article ID 8873639, 6 pages, 2020.

[11] H. Yang, Q. Liu, S. Zhou, and Y. Luo, "A spam filtering method based on multi-modal fusion," Applied Sciences, vol. 9, no. 6, p. 1152, 2019.

[12] P. Ghosh, A. Karim, S. T. Atik, S. Afrin, and M. Saifuzzaman, "Expert cancer model using supervised algorithms with a LASSO selection approach," International Journal of Electrical and Computer Engineering (IJECE), vol. 11, no. 3, p. 2631, 2021.

[13] O. Abayomi-Alli, S. Misra, A. Abayomi-Alli, and M. Odusami, "A review of soft techniques for SMS spam classification: methods, approaches and applications," Engineering Applications of Artificial Intelligence, vol. 86, pp. 197–212, 2019.

[14] M. Bassiouni, M. Ali, and E. A. El-Dahshan, "Ham and spam E-mails classification using machine learning techniques," Journal of Applied Security Research, vol. 13, no. 3, pp. 315–331, 2018.

[15] S. Jeong, G. Noh, H. Oh, and C.-K. Kim, "Follow spam detection based on cascaded social information," Information Sciences, vol. 369, pp. 481–499, 2016.

[16] E. Ezpeleta, U. Zurutuza, and J. M. G. Hidalgo, ''Short messages spam filtering using personality recognition,'' in Proc. 4th Spanish Conf. Inf. Retr., 2016, p. 7.

[17] "SMS Spam Collection Data Set", [Online]. Available: https://archive.ics.uci.edu/ml/datasets/sms+spam+collection. Accessed on – 22-11-2021.

[18] S. Manlangit, S. Azam, B. Shanmugam and A. Karim, "Novel Machine Learning Approach for Analyzing Anonymous Credit Card Fraud Patterns", International Journal of Electronic Commerce Studies 10.2 (2019): 175-202.

[19] S. Zobaed, F. Rabby, I. Hossain, E. Hossain, S. Hasan, A. Karim, and K. Md. Hasib, "Deepfakes: Detecting forged and synthetic media content using machine learning," Advanced Sciences and Technologies for Security Applications, pp. 177–201, 2021.

[20] C. Sitawarin and D. Wagner, "Minimum-norm adversarial examples on KNN and KNN based models," 2020 IEEE Security and Privacy Workshops (SPW), 2020.

[21] B. Wang, Y. K. Yao, X. P. Wang, and X. Y. Chen, "PB-SVM Ensemble: A SVM ensemble algorithm based on SVM," Applied Mechanics and Materials, vol. 701-702, pp. 58–62, 2014.

[22] P. Ghosh et al., "Efficient Predicti on of Cardiovascular Disease Using Machine Learning Algorithms With Relief and LASSO Feature Selection Techniques," in IEEE Access, vol. 9, pp. 19304-19326, 2021, doi: 10.1109/ACCESS.2021.3053759.

[23] P. Ghosh, F. M. Javed Mehedi Shamrat, S. Shultana, S. Afrin, A. A. Anjum and A. A. Khan, "Optimization of Prediction Method of Chronic Kidney Disease Using Machine Learning Algorithm," 2020 15th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP), 2020, pp. 1-6, doi: 10.1109/iSAI-NLP51646.2020.9376787.

[24] Vivekanandam, B. "Design an Adaptive Hybrid Approach for Genetic Algorithm to Detect Effective Malware Detection in Android Division." Journal of Ubiquitous Computing and Communication Technologies 3, no. 2 (2021): 135-149.

[25] P. Ghosh, S. Azam, K. M. Hasib, A. Karim, M. Jonkman and A. Anwar, "A Performance Based Study on Deep Learning Algorithms in the Effective Prediction of Breast Cancer," 2021 International Joint Conference on Neural Networks (IJCNN), 2021, pp. 1-8, doi: 10.1109/IJCNN52387.2021.9534293.

[26] Goswami, Vasudha, Vijay Malviya, and Pratyush Sharma. "Detecting Spam Emails/SMS Using Naive Bayes, Support Vector Machine and Random Forest." In International Conference on Innovative Data Communication Technologies and Application, pp. 608-615. Springer, Cham, 2019.

[27] Dubey, Ratnesh & Mishra, Subha & Choubey, Dilip. "Recognizing Spam Emails/SMS Using Naive Bayes and Support Vector Machine." Complex Systems and Complexity Science Journal, Vol.8 ISSN-NO-1672-3813, 2021.

[28] Prasanna Bharathi, P.,G. Pavani, K. Krishna Varshitha, and Vaddi Radhesyam. "Spam SMS Filtering Using Support Vector Machines." In Intelligent Data Communication Technologies and Internet of Things: Proceedings of ICICI 2020, pp. 653-661. Springer Singapore, 2021.

[29] Rao, Shrihari, and Radhakrishan Gopalapillai. "Effective spam image classification using CNN and transfer learning." In International Conference On Computational Vision and Bio Inspired Computing, pp. 1378-1385. Springer, Cham, 2019.

[30] Manoharan, J. Samuel. "Study of Variants of Extreme Learning Machine (ELM) Brands and its Performance Measure on Classification Algorithm." Journal of Soft Computing Paradigm (JSCP) 3, no. 02 (2021): 83-95.

[31] Haoxiang, Wang, and S. Smys. "A Survey on Digital Fraud Risk Control Management by Automatic Case Management System." Journal of Electrical Engineering and Automation 3, no. 1 (2021): 1-14

[32] Suma, V., and Shavige Malleshwara Hills. "Data Mining based Prediction of Demand in Indian Market for Refurbished Electronics." Journal of Soft Computing Paradigm (JSCP) 2, no. 02 (2020): 101-110

[33] Karthigaikumar, P. "Industrial Quality Prediction System through Data Mining Algorithm." Journal of Electronics and Informatics 3, no. 2 (2021): 126-137.

[34] P. Ghosh, S. Azam, A. Karim, M. Jonkman and MD. Z. Hasan, "Use of Efficient Machine Learning Techniques in the Identification of Patients with Heart Diseases," 5th International Conference on Information System and Data Mining (ICISDM2021), Silicon Valley, USA, pp. 14–20, May 2021, doi: 10.1145/3471287.3471297.