Social Network Analysis and Mining (2023) 13:104
https://doi.org/10.1007/s13278-023-01108-6

ORIGINAL ARTICLE

Machine learning algorithm-based spam detection in social networks

M. Sumathi[1] · **S. P. Raja**[2]

Received: 15 June 2023 / Revised: 24 July 2023 / Accepted: 26 July 2023 / Published online: 19 August 2023

Abstract
Many social media (SM) platforms have emerged as a result of the online social network's (OSN) rapid expansion. SM has become important in day-to-day life, and spammers have turned their attention to SM. Spam detection (SD) is done in two different ways, such as machine learning (ML) and expert-based detection. The expert-based detection technique's accuracy depends on expert knowledge, and it takes huge time to detect the spams. Thus, ML-based spam detection is preferred in OSN. Spam identification on social networks is a difficult operation involving a variety of factors, and spam and ham have resulted in an imbalanced data distribution, which gives flexibility to spammers for corrupting our devices. SD based on ML algorithms like logistic regression (LR), *K*-nearest neighbor (KNN), decision trees (DT), random forest (RF), support vector machine (SVM) and eXtreme gradient boosting (XGB), voting classifier (VC) and extra tree classifier (ETC) are used to design the address balance and to attain high assessment accuracy in an imbalanced datasets. ETC method minimizes the bias through the original sampling process. For reducing processing complexity, the ETC method uses a smaller size constant factor instead of a larger one. Thus, the ETC technique produces better data splitting than DT and RF techniques. Text is vectorized by vectorizers, and all the relative results are stored in it. The VC is an ensemble method that integrates predictions form several methods to forecast an output class depending on which predictions have the highest probability. The multi-class results are aggregated and forecast for the majority voted class. The experimental result shows that, as compared to KN, NB, ETC, RF, SVC, LR, XGB and DT, the proposed VC provides a higher classification accuracy rate of 97.96%, 97.56% of precision, 89.95% of recall and 91.96% of *F*1-measures. Similarly, ETC provides 97.77% accuracy, 98.31% of precision, 84.78% of recall and 91.05% of *F*1-measures. Compared to conventional ML algorithms, VC and ETC provide higher accuracy, precision, recall and *F*1-measures. Thus, ETC and VC are preferable for spam detection. The website has been designed to detect messages as spam or not.

Keywords Social network · Spam features · Spam detection · Machine learning algorithms · Accuracy · Precision · Recall · Voting classifier

1 Introduction

In recent years, the Internet has evolved substantially, and intelligent terminals are becoming increasingly widespread. In this setting, online social networks (OSN) stand out as an essential channel for people to learn, share knowledge, make friends and have fun (Sepideh Bazzaz Abkenar 2021). The OSN's adoption by users, content development, group interactions and information distribution has a significant impact on people's everyday lives, organizational management methods and social stability (Heidemann et al. 2012). This is because of the intricate structure of the OSN, the size of the group and the huge, quick and challenging creation of information that can be tracked (Zhang et al. 2020). The ML models are employed for a variety of purposes across numerous industries. Many people use messages to transfer information, either personal or professional, from one person to another. To spread the spam messages, the spam message link is attached to the original message and sends to the receiver (Janez-Martino et al. 2023). When a spam message link is clicked, the security system of the user is breached

✉ S. P. Raja
avemariaraja@gmail.com

M. Sumathi
sumathishanjai.nitt@gmail.com

[1] School of Computing, SASTRA Deemed University, Thanjavur, Tamilnadu, India

[2] School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, Tamilnadu 632 014, India

by accessing the messaging data and gaining unauthorized access to the user's devices (Vijayaraj et al. 2022).

Many businesses provide SD technology and methods. By using those technology and methods, there were several spam messages filtered. Several companies, including Google, Outlook and Hey, have shown significant success in the detection of spam communications (Mateen 2017). A variety of filtering techniques are used to prevent the identification of spam communications because ML models can be trained to independently detect spam and legitimate messages and test them with new messages. Hence, ML-based detection is an easiest way to detect the spams (Chakraborty et al. 2016). A variety of performance metrics need to be utilized to classify the communications as spam or ham. Different performance metrics lead to various best-suited ML models. In addition to ML models, there are other methods that can be used to identify spam communications. To improve the understanding of the outcomes, integrated ML models are required (Madisetty and Desarkar 2018). As a result, the website and many ML classifiers to identify spam and ham messages have been developed in this work. The spam and ham models have been identified as the most appropriate models after comparing the findings and evaluating their performance metrics. An efficient method is provided for testing our findings with user input utilizing a few tools (Stringhini et al. 2010).

The merits of ML algorithms in spam detection are that the unsupervised ML algorithm is particularly beneficial for real-time unlabeled data. The best model has been found after comparing the accuracy ratings of various ML classifiers (Govil et al. 2020b). The findings had been compared by using a variety of performance criteria, and each one's analysis yields a different optimal technique. Instead of utilizing a random approach to identify the spam, the ML algorithms help to classify the spam in the best way (Zheng et al. 2016). The ML algorithms demonstrate the optimal model for the dataset in ETC and VC. It may also indicate the amount of time needed to train and test various ML algorithms. The split sets are used to show how results change as the ratio of training to testing sets changes. Appropriate performance measures are used to assess the effectiveness of the various ML classifiers (Choi and Jeon 2021).

Every technique has some demerits also. The demerits of the ML algorithms is, hard to find if it the values are "overfitted." The performance measures that are derived for a suitable model need to be more specific. Initializing the settings is a laborious process. There has to be further fine-tuning (Swathi 2018). The ML algorithms take longer to achieve optimal performance since more datasets are needed for training to produce results that are more accurate (Hu et al. 2014). Even though they do not always produce the best results, some classifiers, like SVM, require more time for training and validating the data. It does not provide the most accurate results when allowing real-time user inputs. The model selection process would become arbitrary, and factories revealed that this was frequently unsatisfactory (Sharma and Kaur 2016). Unsupervised ML algorithms typically fail due to the large number of subjective judgments required to even get them to work, resulting in poor quality, difficult-to-understand models that cannot be argued. It requires more talent, human adjustment and feedback when compared to supervised learning projects to create value from subject matter experts (Ahmed and MAbulaish 2013).

To overcome the aforementioned demerits, the major contribution of the proposed work is as follows:

- To analyze the existing works related to ML-based SD in a detailed manner.
- To collect the spam dataset for performing classification of both spam and ham.
- To provide the user information regarding relevant and false messages.
- To determine whether or not the communication is spam.

The remaining part of this article is organized as follows: In the literature review section, the merits and demerits of the existing works related to SD by using ML techniques are discussed in detail. In the proposed methodology, the proposed voting classification technique is discussed with system architecture and necessary algorithms. In the experimental results section, the proposed technique result is compared to existing spam detection techniques. Finally, the proposed system is concluded with future enhancements.

2 Literature review

Nikhil Govil et al. proposed the ML-based SD mechanism for preventing various phishing attacks through dictionary generation. After generating the dictionary, the features had generated by using ML algorithms. Afterward, the generated features have been tested thoroughly and passed to the NB algorithm. The NB algorithm calculated the probability rate of the e-mails and classified them as spam or ham. Compared to other ML algorithms, the NB gave low performance and had worked well for e-mail-based SD (Govil et al. 2020a). Gupta et al. studied SD in short message services (SMS) by using ML algorithms. The deep learning-based convolutional neural network (CNN) works better than the SVM and NB algorithms. Likewise, the image-based SD has been done through the CNN technique. This technique worked well for some smaller datasets and not for large datasets (Gupta et al. 2018). Masood et al. detect spam and fake users on the social network. The malware alerting system and regression prediction models were used for the fake content prediction. The Twitter content was analyzed

to identify fake content and users, spam in the URL's and trending topics. This work analyzed in detail the prevention of fake accounts and the spread of fake news. Fake news and user predictions were extremely difficult to process when dealing with large amounts of media data (Masood et al. 2019).

Jbara et al. proposed SD in Twitter using an URL-based detection technique. Nowadays, spammers are the major platform to demand social networks and spread irrelevant data to users. In particular, Twitter is the most prominent network to spread spam among the social networks. To avoid this spread, the author used URL- and ML-based detection techniques. Compared to other ML algorithms, the RF-based classification technique provided a higher accuracy rate of 99.2%. In this work, 70% data were used as training data and 30% data were used for testing purposes (Jbara and Mohamed 2020). Asif Karim et al. surveyed the state of intelligent SD in e-mail. Both artificial intelligence and ML methods were used for intelligent SD. This combined approach protected e-mails from phishing attacks. Apart from content filtering, the other methods have been covered in lesser percentage in this analysis (Karim et al. 2019). Huang et al. proposed the regression and multi-class classification-based extreme learning techniques for SD. It is shown that both the learning framework of SVM and extreme learning machines (ELM) can be implemented. It has provided better scalability and faster learning speed. But it has provided very low performance rate (Huang et al. 2012).

Zhao et al. discussed the ensemble learning-based SD with imbalanced data in social networks. The heterogeneous-based ensemble technique had been used in the imbalance class to detect spam in OSN. The base and combine modules were integrated for finding spam in an OSN. In the base module, the basic ML algorithms were used to find the spam, and in the combine module, the deep learning-based neural network was used for SD with dynamic adjustment of weight values. This technique works well for Twitter-based real spam datasets but not for hidden features (Zhao et al. 2020). Gauri Jain et al. proposed the convolutional and long short-term memory-based neural network (LSTM) technique for SD. The CNN and LSTM were combined to detect spam on the Twitter network. The knowledge-based technique was used to improve the prediction accuracy of SD. This technique had been works well on short messages like Twitter messages instead of lengthy e-mail messages (Jain et al. 2019). Barushka et al. discussed the cost-sensitive and ensemble-based deep neural networks for SD on OSN. Traditional ML algorithms, such as SVM and NB techniques, are unsuitable for high-dimensional data on OSN. To reduce the misclassification cost and the number of attributes in the spam filtering process, the multi-objective evolutionary feature selection process was used in this work. The deep neural network and cost-sensitive learners were used to regularize the learning process (Barushka and Hajek 2020).

Pirozmand et al. used the force-based heuristic algorithm for OSN SD. The ML- and deep learning-based integrated technique was used for spam filtering in OSN. The SVM, genetic algorithm (GA) and gravitational emulation local search (GELS) algorithm were integrated to filter spam in OSN. This integrated technique selects the highly effective features of the spam filter. The enhanced GA helped to select the feature based on exploration, and GELS helped to improve exploration and local search. To improve the detection accuracy, several levels of modifications were made in the algorithm (Pirozmand and Sadeghilalimi 2021). Zheng et al. discussed the SD on social networks. The dataset was constructed with more than 16 million labeled messages. Afterward, a manual classification was performed to classify the spam and ham data. Then the user's behavior and message content were extracted from the social network for applying the SVM algorithm. This technique provided more than 99.9% accuracy than the other algorithms. In this technique, the computational complexity of manual processes is very high (Zheng et al. 2015). Alom et al. proposed the deep learning model to SD on Twitter. Generally, ML algorithms are used for SD in most of the applications. But the ML algorithms have not been work well on OSN. Hence, the deep learning algorithm was proposed by the author to filter the spam. The tweet text and user meta-data were analyzed to detect the spam. Compared to basic ML algorithms, the deep learning algorithms provided better results (Alom et al. 2020). Table 1 shows the ML-based SD.

3 Summary of the existing work

Based on the above literature review, the following challenges are identified in the conventional SD techniques.

- The conventional ML algorithms are works well for lesser sized data not effective to larger sized data.
- Fake news and user predictions were extremely difficult to process when dealing with large amounts of media data.
- Some ML algorithms support high scalability but lower in performance rate.
- Deep neural networks work well in explicit data not for hidden features.
- Ensemble technique works fine for shorter message and to lengthy messages like e-mails.
- Compared to ML algorithms, the deep learning algorithms are working well to detect spam. But, the computational complexity of deep learning is higher than ML algorithms.

Table 1 Works related to ML-based SD

Author and paper details	Technique and parameter used	Dataset used	Merits	Limitations
Niranjani et al. (2022) SD in OSN using ML	Natural language-based term identification and sentiment classification. mean square error, root mean absolute error	Twitter, Facebook and Instagram	Less mean absolute error and mean square error	Higher processing time than ML algorithms
Elakkiya et al. (2022) SD using feed-forward neural network	Reinforcement learning and *k*-norm factor-based shuffled frog leaping algorithm. Accuracy, MSE, AUC and detection rate	Tip spam dataset and Twitter dataset	Higher accuracy rate and lower false positive rate	Works well to numerical data not to text-based messages
Gradhi Svadasu et al. (2022) SD using artificial neural network (ANN)	Artificial neural network and SVM, KNN. Accuracy, precision and *F*1-score	Social media dataset	Compared to SVM, KNN has produced less error rate and ANN produce better results than KNN and SVM	Limited features were considered for SD and higher detection time is required due to algorithm complexity
Jenifer et al. (2022) Multi-objective and CNN-based SD	MOGA-CNN-DLAS. Accuracy, precision, recall, *F*1-Score, RMSE and MAE	Twitter 100 k and ASU dataset	High accuracy, precision, recall, *F*1-score, mean absolute error (MAE) and root mean square error (RMSE)	Focused on single objective not to multi-objective optimization
Anisha et al. (2022) ML- and deep learning-based SD	Multinomial NB and LSTM. Accuracy, *F*1-Score, recall and precision	Real-time Twitter dataset	Compared to multinomial NB results, the LSTM accuracy rate is high	Individual account-based SD is performed not to followers
Naeem Ahmed et al. (2022) SD in e-mail	NB, DT, NN, RF, SVM. Accuracy, precision and recall	E-mail dataset	Compared to DT, NN and RF, NB and SVM outperform	Analyzed different types of supervised ML algorithms only
Deepjyoti Choudhury et al. GA-based fake news detection (Choudhury and Acharjee 2022)	Genetic algorithm. Precision, recall, *F*1-Score and Accuracy	Fake job posting and Fake news dataset	SVM and LR classifiers outperform in fake news dataset	Support minimal number of features not for larger features
Nan Sun et al. (2022) SD in near real-time Twitter data	Parallel computing	Twitter dataset	RF outperforms for near real-time Twitter dataset	Support limited data size
Merly Thomas et al. (2023) SD in twitter data using Feature fusion	Feature fusion and fuzzy network. Precision (0.894), recall (0.903) and *F*-measure (0.898)	5 k continuous Twitter dataset	Effectively working in high-dimensional data in real environment	Another deep learning framework is required to increase spam detection efficiency
Chanchal Kumar et al. (2023) A hybrid data-driven SD in OSN	Sampling algorithm combined with edited nearest neighbors (SMOTE-ENN). Accuracy (99.26), precision (99.07), recall (99.49)	Random Unbalanced Twitter Dataset	Random forest produced higher precision rate than DT, KNN, Gaussian Naïve Bayes	Algorithm efficiency is lower than deep learning and ML algorithms
Venkateswarlu et al. (2021) feature fusion-based SD	Renyi entropy and deep belief network. Precision (97.3%), recall (99.2%) and *F*-measure (98.2%)	Twitter Data	Using data transmission process, the feature reordering and grouping is easiest.	Required higher storage space than other techniques

3.1 Contribution of the proposed work

Based on the above analysis, ML algorithms identify the spam in lesser complexity, but the accuracy depends on the dataset and type of ML algorithm used for SD. In most of the analysis, RF, SVM, NB and CNN outperform than the other classification techniques. To improve the prediction accuracy, the alternate technique is required in current scenario. Thus, the ML-based voting classifier is proposed in this work for classifying spam and ham. Two different imbalanced datasets are used in the proposed work. One dataset collected from Kaggle dataset and another from nsclab resources.

4 Proposed methodology

In this section, the proposed voting classification-based SD technique is discussed with the necessary architecture and algorithms.

1. *Dataset (D)*: A dataset is a group of connected pieces of information or data that are put together for a specific element. The dataset is obtained from Kaggle (https://raw.githubusercontent.com/mohitgupta-omg/Kaggle-SMS-Spam-Collection-Dataset-/master/spam.csv), which provides the dataset for training the models with 5500 + data messages. In the present work, two attributes named "target" and "text" are used for processing. The target column tells whether the text corresponding to it is ham or spam. The text column contains text which includes both ham and spam messages. The Twitter spam dataset is used for imbalanced data processing (Zhao et al. 2020). The Twitter4J library and Twitter API are used for Tweets collection process which contains 600 million tweets and 6.5 million malicious tweets. Another imbalanced dataset collected from http://nsclab.org/nsclab/resources/. This dataset contains 5 k random and continuous data, and 95 k random and continuous data. Twelve attributes are involved in this dataset such as age, lists, following, number of follower, tweets, user favorites, retweets, URL's, number of digits, user mention and hashtag.
2. *Data cleaning and preprocessing*: In data cleaning, the removal of unnamed columns, renaming the columns, finding the missing values, checking for duplicate values and removing the duplicate values have been carried out. Label encoding is used to encode the text to binary values 1 and 0, which represents spam as 0 and ham as 1. In data preprocessing, conversion to lowercase, tokenization and removal of special characters, commas, punctuation and stop words are carried out. After that, stemming process has applied on it. After that, all alphanumeric words are processed into another column. These numerical values act as an input data.
3. *Data splitting*: Data splitting is the process of dividing data into training and testing sets. The imported function train_test_split is used to divide the data collection into training and testing data. Four arrays, i.e., *Y* Train, *Y* Test, *X* Test and *X* Train, are utilized to do the splitting of data. 80% of the data from the original dataset is used for training, and the remaining 20% is used for testing.
4. *Model building*: DT, SVM, RF, KNN, LR, XGB and voting classifier are tested, and metrics such as accuracy and precision are calculated. Accuracy comparison and cross-validating the results have been carried out in the existing and proposed algorithm.
5. *Support vector machine*: In SVM, the cluster of data is divided into its appropriate groups by a hyperplane using a classification strategy, which shows every node in a dimensional plane that comes from a dataset. This approach optimizes the linear algorithm by iterating over sample data using the learning rate. The major advantages of SVM over other ML algorithms are: run faster and performs well on a minimal dataset. When a dataset size is larger, SVM processes the data at lower level, and afterward converts it to a higher level. SVM works well for SD in the minimal dataset.
6. *Decision tree classifier*: The DT model is constructed using the predictive approach. The algorithm continues until either the user exits or the software reaches its end decision. By using the training data, this model learns to predict the value of the data. The accuracy rate of the DT depends on the extensiveness and deeper of the tree and the more complex the set of rules are followed in the classification. In DT, features are represented in internal nodes, decision rules are represented in braches, and the results are produced in leaf nodes. The decision node helps to make a decision through branches and leaf node produces the outcome of each decision. Equation 1 is used to find the decision in DT.

$$H(s) = \left(-\text{Prob}\left(\log_2(p+)\right)\right) - \left(-\text{Prob}(\log_2(p-))\right) \quad (1)$$

where $(p+)$ is the percentage of the positive class and $(p-)$ is the negative class. Figure 1 shows the working flow of the proposed system.

4.1 Extra tree classifier (ETC)

The ETC algorithm is quite similar to the DT and RF techniques for selecting the victim attributes. By combining the output of numerous DT, a forest is created to print the outcome. The initial training dataset produced the additional tree. For each test case, the ETC selects the optimally best

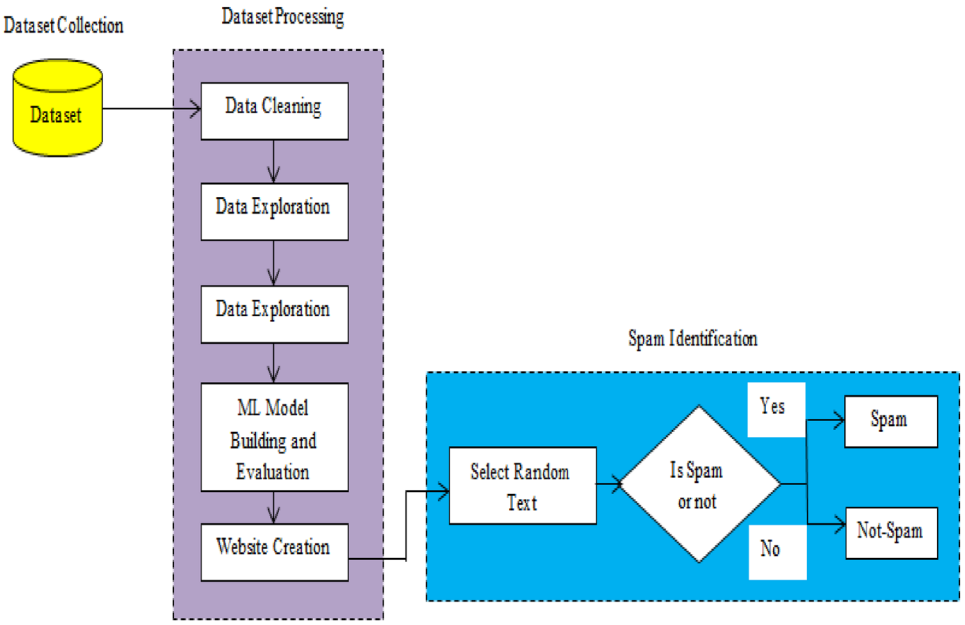

Fig. 1 Flow diagram of SD

attribute by a Gini Index. Equation 2 is used to find the Gini index value of an attribute.

$$\text{Gini}_{\{\text{index}\}} = 1 - \sum_{i=1}^{C} (p_i)^2 \tag{2}$$

where "*c*" represents the total number of unique classes. Algorithm 1 is used for performing the ETC process for splitting the data features.

Compared to the ensemble technique, the ETC method minimizes the bias through the original sampling process. To reduce processing complexity, the ETC method uses a smaller size constant factor instead of a larger one. Thus, the ETC technique produces better data splitting than DT and RF techniques.

Algorithm 1: Extra Tree Classification – Data Split

Input : Dataset D, Gini Random Function (K)
Output: Split Data
Procedure:

1. Select K number of features from D
2. Conduct split on D using split(D, K)
3. In split(D, K) – Select minimum (X_{min}) and maximum (X_{max}) values
4. Find the cut-point K_c randomly using X_{min} and X_{max}
5. Make a split using [$K < K_c$]
6. If |M| < Xmin, return true
7. All features in X are equal, return true
8. Else return false
9. End if

4.2 Voting classifier (VC)

The VC is an ensemble method that integrates predictions from several models to forecast an output class depending on which predictions have the highest probability. The voting classifier method just adds up the results of each classifier that were fed into the model and predicts the output class depending on which class received the most votes, such that multiple class results are aggregated and forecast for the majority voted class. Algorithm 2 shows the VC process in the proposed work. Based on algorithm 2, the testing data are classified as spam or ham.

Table 2 Performance measures

Parameter	Measures	
	Spam	Ham
Spam	TP	FN
Actual non-spam	FP	TN

Algorithm 2: Voting Classifier Based SD

Input: Output of classifiers
Output: Spam or Ham
Procedure:

1. Split data as Training and Testing using ETC.
2. Returning training and Testing data
3. If voting = "soft"
   M_1=DT(TN_data, TT_data, TT_label) // TN_data – Training data, TT_data – Testing data
   M_2=LR(TN_data, TT_data, TT_label) // TN_label – Training Label
   M_3=RF(TN_data, TT_data, TT_label)
   M_4=SVM(TN_data, TT_data, TT_label)
   M_5=NB(TN_data, TT_data, TT_label)
   M_6=XGB(TN_data, TT_data, TT_label)
   M_7=KN(TN_data, TT_data, TT_label)
   M_8=ETC(TN_data, TT_data, TT_label)
   M_9=VC(TN_data, TT_data, TT_label)
4. Procedure Ensemble(TN_data, TT_data, TT_label)
5. Soft_VC=Concatenate(M_1,M_2,M_3,M_4,M_5,M_6,M_7,M_8,M_9)
6. Soft_VC.fit(TN_data, TN_label)
7. Predictions=soft_VC.predict(TT_data)

4.3 Website development

Using the developed website, a random text is predicted whether it will be "spam" or "ham." Visual Studio code is used to execute this website. An open-source Python toolkit called Streamlet makes it simple to develop and distribute stunning, personalized web apps for data science and machine learning.

4.4 Calculating the performance measures

The values of false positives (FP) and negatives (FN), as well as true positives (TP) and true negatives (TN), are provided by the matrix. The accuracy, precision and recall scores are calculated using these matrix values. The *F*1-score can be calculated using precision and recall values. The following equations are used for finding the accuracy, precision, recall

Fig. 2 Initial dataset

	v1	v2	Unnamed: 2	Unnamed: 3	Unnamed: 4
4910	ham	Love that holiday Monday feeling even if I hav...	NaN	NaN	NaN
2761	ham	I am not sure about night menu. . . I know onl...	NaN	NaN	NaN
3621	ham	Goin to workout lor... Muz lose e fats...	NaN	NaN	NaN
1576	ham	No. To be nosy I guess. Idk am I over reacting...	NaN	NaN	NaN
5370	spam	dating:i have had two of these. Only started a...	NaN	NaN	NaN

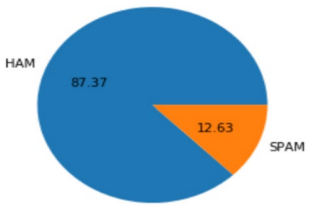

Fig. 3 Spam and ham ratio in initial dataset

Table 3 Data frame information

Sl. No	Column name	Non-null count	Data type
0	*V1*	5572	Object
1	*V*2	5572	Object
2	Unnamed: 2	50	Object
3	Unnamed: 3	12	Object
4	Unnamed: 4	6	Object

and *F*1-score values of the proposed system (Sepideh Bazzaz Abkenar 2021). Table 2 shows the performance measures of the proposed system.

$$\text{Accuracy} = \frac{(\text{TN} + \text{TP})}{(\text{TP} + \text{FN} + \text{FP} + \text{TN})} \tag{3}$$

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

$$\text{Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \tag{5}$$

$$F1 = 2 * \frac{(\text{Precision} * \text{Recall})}{(\text{Precision} + \text{Recall})} \tag{6}$$

where TP = true positive, which is a spam message anticipated to be spam, and TN = true negative, which is a ham message predicted to be ham. Ham messages were mistakenly identified as spam (FP), and spam messages were mistakenly identified as ham (FN).

5 Experimental results and discussion

The proposed system is implemented on the Windows 10 operating system with the Python language, 8 GB of RAM and a 2.40 GHz CPU. The Jypyter Notebook and Visual Studio Code are used for website development. In a proposed system, 5500 + data messages are analyzed for spam and ham detection. The training and testing datasets are split into 80:20 ratios for balanced dataset. The randomly selected 50% samples are used for training, and the remaining 50% is used for testing of imbalanced dataset.

5.1 Dataset description

Figure 2 shows the initial dataset of the proposed system. The dataset contains 5 columns, such as number of records, type of data (spam or ham), testing message and three unlabeled attributes. The dataset was evaluated by different ML

Fig. 4 Dataset after preprocessing

	target	text	num_characters	num_words	num_sentences	transformed_text
0	0	Go until jurong point, crazy.. Available only ...	111	23	2	go jurong point avail bugi n great world la e ...
1	0	Ok lar... Joking wif u oni...	29	8	2	ok lar joke wif u oni
2	1	Free entry in 2 a wkly comp to win FA Cup fina...	155	37	2	free entri 2 wkli comp win fa cup final tkt 21...
3	0	U dun say so early hor... U c already then say...	49	13	1	u dun say earli hor u c alreadi say
4	0	Nah I don't think he goes to usf, he lives aro...	61	15	1	nah think goe usf live around though

	num_characters	num_words	num_sentences
count	5169.000000	5169.000000	5169.000000
mean	78.977945	18.286903	1.961308
std	58.236293	13.227173	1.432583
min	2.000000	1.000000	1.000000
25%	36.000000	9.000000	1.000000
50%	60.000000	15.000000	1.000000
75%	117.000000	26.000000	2.000000
max	910.000000	219.000000	38.000000

(a) Total number of Messages

	target	text	num_characters	num_words	num_sentences
0	0	Go until jurong point, crazy.. Available only ...	111	23	2
1	0	Ok lar... Joking wif u oni...	29	8	2
2	1	Free entry in 2 a wkly comp to win FA Cup fina...	155	37	2
3	0	U dun say so early hor... U c already then say...	49	13	1
4	0	Nah I don't think he goes to usf, he lives aro...	61	15	1

(b) Type of Spam Messages

	num_characters	num_words	num_sentences
count	653.000000	653.000000	653.000000
mean	137.891271	27.474732	2.969372
std	30.137753	6.893007	1.488910
min	13.000000	2.000000	1.000000
25%	132.000000	25.000000	2.000000
50%	149.000000	29.000000	3.000000
75%	157.000000	32.000000	4.000000
max	224.000000	44.000000	9.000000

(c) Number of Character, words and sentences in spam messages

	num_characters	num_words	num_sentences
count	4516.000000	4516.000000	4516.000000
mean	70.459256	16.958370	1.815545
std	56.358207	13.395014	1.364098
min	2.000000	1.000000	1.000000
25%	34.000000	8.000000	1.000000
50%	52.000000	13.000000	1.000000
75%	90.000000	22.000000	2.000000
max	910.000000	219.000000	38.000000

(d) Number of Character, words and sentences in ham messages

Fig. 5 **a** to **d** Exploratory data analysis

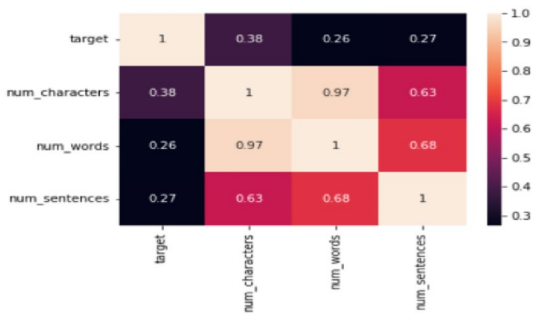

Fig. 6 Correlation between columns

algorithms like KN, NB, ETC, RF, SVC, LR, XGB and DT. These algorithm performances are compared to the proposed VC algorithm performance in terms of accuracy, precision, recall and *F*1-measures. Accuracy of the proposed system is measured by the correctly identified spam from the total dataset. Figure 3 shows the spam and ham ratio of input dataset with 5500 + messages with 87.37% ham and 12.63% spam.

Table 3 shows the data frame details of the dataset, such as the number of values in each attribute and its data type. Both *V*1 and *V*2 have the associated values for the further process. These data values are applied to different ML algorithms to find the accuracy rate of each algorithm. Now preprocessing is applied to the dataset to identify the required attributes for spam detection.

After preprocessing, the dataset contains the actual information, which is required for SD. Figure 4 shows the dataset after preprocessing consists of data with spam and ham messages.

In exploratory data analysis (EDA), the duplicated instances, nulls and missing instances are eliminated. In a proposed dataset, after eliminating 403 duplicated messages, 5159 messages are identified as non-duplicated messages. Following that, 653 messages are classified as spam, while the remaining messages are classified as ham.

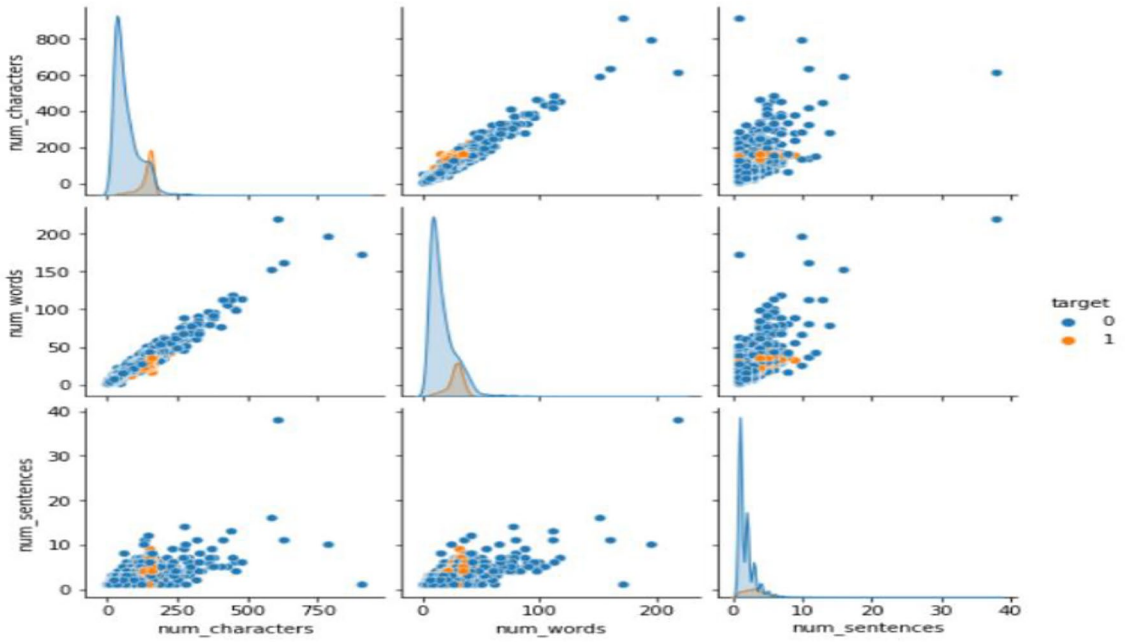

Fig. 7 Pair plot representation of the attributes

Table 4 Performance comparison of ML algorithms

Algorithm	Accuracy	Precision	Recall	*F*1-measure
KN	0.905222	1.000000	0.289855	0.449438
NB	0.972921	1.000000	0.797101	0.887097
ETC	0.977756	0.983193	0.847826	0.910506
RF	0.971954	0.973913	0.811594	0.885375
SVM	0.974855	0.966667	0.840580	0.899225
LR	0.957447	0.951923	0.717391	0.818182
XGB	0.969052	0.941667	0.818841	0.875969
DT	0.935203	0.858586	0.615942	0.717300
VC	**0.979691**	**0.975610**	**0.869565**	**0.919540**

Bold terms shows that the proposed VC technique provided improved results than the other methods is proven

The dataset message also specifies the number of characters, sentences and words. Figure 5a–d shows the number of characters, words and sentences in the total messages, spam messages and ham messages.

5.2 Correlation of the columns

Figure 6 shows the correlation relationship between columns present in the dataset. The number of characters–words relationship has the highest frequency value of 0.38. This shows that the number of characters and their related words play a vital role in identifying spam messages. The remaining factor-based correlations like number of characters–sentences, word–character, sentence–character and sentence–word are somewhat lower

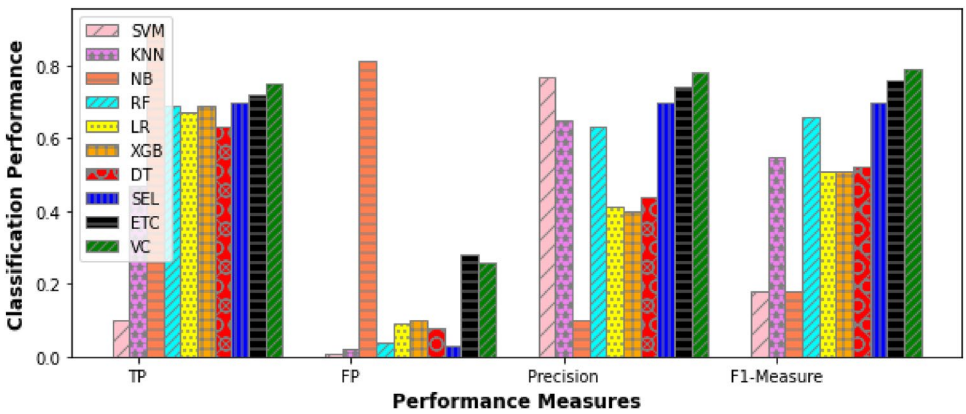

Fig. 8 Imbalance dataset comparison with different ML algorithms

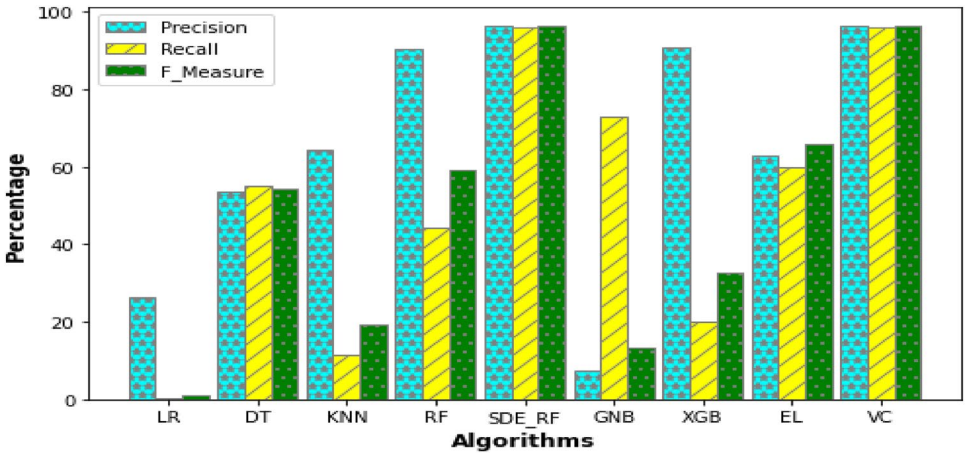

Fig. 9 Imbalance dataset comparison to other algorithms

frequency values like 0.26 and 0.27. In such cases, the words and sentences are also helpful in identifying the spam messages. Thus, SD is mainly focused on character–word frequency analysis.

5.3 Pair plot between the columns

The objective of the proposed work is to identify the spam messages from the dataset. The pair plot is used to determine the relationship between columns such as character count, words and sentences. Based on these relationships, the spam message range is easily identified in the dataset. The number of sentences beyond 10, the number of words beyond 50 and the number of characters beyond 200 all contain more spam messages. Figure 7 shows the pair plot representation of the attributes present in the dataset.

5.4 Performance comparison of ML algorithms

Different ML algorithms like KN, NB, ETC, RF, SVC, LR, XGB, DT and VC are executed in the preprocessed dataset and measured for accuracy, precision, recall and *F*1-measure. Table 4 shows the accuracy, precision, recall and *F*1-measures of each algorithm. Based on the accuracy analysis, VC has produced a higher accuracy rate of 97.96%, 97.56% for precision, 86.95% for recall and 91.96% for *F*1-measure. Afterward, ETC provides the next level of accuracy as 97.77%, 98.31% of precision, 84.78% of recall and 91.05% of *F*1-measures. Compared to conventional ML algorithms, VC and ETC provide higher accuracy, precision, recall and *F*1-measures. Thus, ETC and VC are preferable for SD.

5.5 Imbalanced dataset classification analysis

The Twitter dataset is considered for the imbalanced dataset classification process. Various proportion rates like 50:50 are considered for analysis. The precision, recall, accuracy and *F*1-measures are considered for the ML algorithms processing. Figure 8 shows the comparison of different ML algorithms with different ratio. The performance of SVM, NB, KN, RF, LR, XGB, DT, stacking-based ensemble learning (SEL) and ETC is compared to the proposed VC technique. The proposed VC technique provides better detection rate than the other algorithms on the imbalanced dataset has been proven in the results.

The imbalanced dataset contains 1:19 ratio of spam and non-spam data. Two different types of data are considered for the analysis such as randomly gathered data and continuous data. The proposed work is compared to Sepideh Bazzaz Abkenar (2021) and Zhao et al. (2020). Both approaches used the same dataset for the classification of spam and non-spam. Figure 9 shows the comparison of proposed work, basic classifiers (Sepideh Bazzaz Abkenar 2021; Zhao et al. 2020).

When compared to SDE_RF technique, the proposed VC technique result is improved in 0.05%. It has been shown in the above-mentioned graph.

6 Conclusion

The proposed spam detection technique classifies the spam and ham messages by using ETC and VC algorithms. The ETC algorithms split the data in an accurate manner by

combining the output of numerous DT. The ETC is created to print the outcome and initial training dataset produced the additional tree. In VC, to produce higher probability prediction results, several methods are integrated into single model. The VC technique adds the results of each classifier and predicts the output class depending on which class received the most votes. VC has produced a higher accuracy rate of 97.96%, 97.56% for precision, 86.95% for recall and 91.96% for $F1$-measure. Afterward, ETC provides the next level of accuracy as 97.77%, 98.31% of precision, 84.78% of recall and 91.05% of $F1$-measures. Compared to conventional ML algorithms, VC and ETC provide higher accuracy, precision, recall and $F1$-measures. Thus, ETC and VC are preferable for SD. The training and testing datasets are created from the source dataset based on the examination of the experiential results. Finally, the accuracy, precision, recall and $F1$-Score are predicted using classification-based machine learning algorithms. Because of the great results, the VC algorithm efficiently classified the messages as spam and ham. Then, the ETC model's almost perfect specificity successfully identified the ham signals. ETC also demonstrates that spam messaging capabilities are good. To obtain even greater performance in the future, it may be conceivable to add modifications or enhancements to the suggested system and classification algorithms. Future developments will see the stacking ensemble architecture and apply our methodology to other real-world applications. To improve accuracy, the Gaussian mixture model (GMM) will be proposed in future work.

Authors contributions MS had done the methodology. SPR had done the writing and drafting. All the authors are aware of the submission.

Funding Not applicable.

Data availability Data will be made available based on the request.

Declarations

Conflict of interest We declare that there is no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

Abkenar SB, Kashani MH, Mahdipour E, Jameii SM (2021) Big data analytics meets social media: a systematic review of techniques, open issues, and future directions. Telematics Inf 57:101517

Ahmed F, Abulaish M (2013) A generic statistical approach for spam detection in online social networks. Comput Commun 36(10–11):1120–1129

Ahmed N, Amin R, Aldabbas H, Koundal D (2022) Machine learning techniques for spam detection in Email and IoT platforms: analysis and research challenges. Secur Commun Netw 8:1–19

Alom Z, Carminati B, Ferrari E (2020) A deep learning model for Twitter spam detection. Online Soc Netw Media 18:1–12

Barushka A, Hajek P (2020) Spam detection on social networks using cost-sensitive feature selection and ensemble-based regularized deep neural networks. Neural Comput Appl 32:1–19

Chakraborty M, Pal S, Pramanik R, Ravindranath Chowdary C (2016) Recent developments in social spam detection and combating techniques: a survey. Inf Process Manag 52(6):1053–1073

Choi J, Jeon C (2021) Cost-based heterogeneous learning framework for real-time spam detection in social networks with expert decisions. IEEE Access 9:103573–103587

Choudhury D, Acharjee T (2022) A novel approach to fake news detection in social networks using genetic algorithm applying machine learning classifiers. Multimed Tools Appl 82:1–17

Elakkiya E, Selvakumar S (2022) Stratified hyperparameters optimization of feed-forward neural network for social network spam detection (SON2S). Soft Comput 8:1–20

Govil N, Agarwal K, Bansal A, Varshney A (2020a) A machine learning based spam detection mechanism. In: Fourth international conference on computing methodologies and communication (ICCMC 2020a), pp 954–957

Govil N, Agarwal K, Bansal A, Varshney A (2020b) A machine learning based spam detection mechanism. In: 2020b Fourth international conference on computing methodologies and communication (ICCMC), Erode, India

Gupta M, Bakliwal A, Agarwal S, Mehndiratta P (2018) A comparative study of spam SMS detection using machine learning classifiers. In: 2018 Eleventh international conference on contemporary computing (IC3), pp 1–7

Heidemann J, Klier M, Probst F (2012) Online social networks: a survey of a global phenomenon. Comput Netw 56:3866–3878

Hu X, Tang J, Liu H (2014) Online social spammer detection. In: Proceeding 28th AAAI conference on artificial intelligence (AAAI), pp 59–65

Huang G-B, Zhou H, Ding X, Zhang R (2012) Extreme learning machine for regression and multiclass classification. IEEE Trans Syst Man Cybern 42(2):513–529

Jain G, Sharma M, Agarwal B (2019) Spam detection in social media using convolutional and long short term memory neural network. Ann Math Artif Intell 85:21–44

Janez-Martino F, Alaiz-Rodriguez R, Gonzalez-Castro V, Fidalgo E, Alegre E (2023) A review of spam email detection: analysis of spammer strategies and the dataset shift problem. Artif Intell Rev 56:1145–1173

Jbara YHF, Mohamed HAS (2020) Twitter spammer identification using URL based detection. IOP Conf Ser Mater Sci Eng 925:1–7

Jenifer Darling Rosita P, Jacob WS (2022) Multi-objective genetic algorithm and CNN-based deep learning architectural scheme for effective spam detection. Int J Intell Netw 3:9–15

Karim A, Azam S, Shanmugam B, Kannoorpatti K, Alazab M (2019) A comprehensive survey for intelligent spam email detection. IEEE Access 7:168261–168295

Kumar C, Bharti TS, Prakash S (2023) A hybrid data-driven framework for spam detection in online social network. In: International conference of machine learning and data engineering

Madisetty S, Desarkar MS (2018) A neural network-based ensemble approach for spam detection in twitter. IEEE Trans Comput Soc Syst 5(4):973–984

Masood F, Ammad G, Almogren A, Abbas A (2019) Spammer detection and fake user identification on social networks. IEEE Access 7:68140–68152

Mateen M, Iqbal MA, Aleem M, Islam MA (2017) A hybrid approach for spam detection for Twitter. In: 2017 14th International Bhurban conference on applied sciences and technology (IBCAST), Islamabad, Pakistan, pp 466–471

Niranjani V, Agalya Y, Charunandhini K, Gayathri K, Gayathri R (2022) Spam detection for social media networks using machine learning. In:2022 8th International conference on advanced computing and communication systems (ICACCS), pp 2082–2088

Pirozmand P, Sadeghilalimi M, Rahmani AA (2021) A feature selection approach for spam detection in social networks using gravitational force-based heuristic algorithm. J Amb Intell Human Comput 8:1–14

Rodrigues AP, Fernandes R, Aakash A, Abhishek B, Shetty A (2022) Real-time twitter spam detection and sentiment analysis using machine learning and deep learning techniques. Comput Intell Neurosci 2022:1–14

Sharma R, Kaur G (2016) E-mail spam detection using SVM and RBF. Int J Mod Educ Comput Sci 8:57–63

Stringhini G, Kruegel C, Vigna G (2010) Detecting spammers on social networks. In Proceeding 26th annual computer security application conference (ACSAC), pp 1–9

Sun N, Lin G, Qiu J, Rimba P (2022) Near real-time twitter spam detection with machine learning techniques. Int J Comput Appl 44:1–12

Svadasu G, Adimoolam M (2022) Spam detection in social media using artificial neural network algorithm and comparing accuracy with support vector machine algorithm. In: 2022 International conference on business analytics for technology and security (ICBATS), pp 1–5

Swathi P (2018) Analysis on solutions for over-fitting and under-fitting in machine learning algorithms. Int J Innov Res Sci Eng Technol 7:10–15680

Thomas M, Meshram BB (2023) Chso-DNFNet: spam detection in Twitter using feature fusion and optimized deep neuro fuzzy network. Adv Eng Softw 175:1–12

Venkatewarlu B, Viswanath Shenoi V (2021) Optimized generative adversarial network with fractional calculus based feature fusion using twitter stream for spam detection. Inf Secur J Glob Perspect 8:1–20

Vijayaraj N, Sumathi M, Rajkamal MU (2022) Decision trees to detect malware in a cloud computing environment. In: 2022 International conference on electronic systems and intelligent computing (ICESIC), pp 299–303

Zhang Z, Hou R, Yang J (2020) Detection of social network spam based on improved extreme learning machine. IEEE Access 8:112003–112014

Zhao C, Xin Y, Li X, Yang Y, Chen Y (2020) A heterogeneous ensemble learning framework for spam detection in social networks with imbalanced data. Appl Sci 10:1–18

Zheng X, Zeng Z, Chen Z, Yuanlong Yu, Rong C (2015) Detecting spammers on social networks. Neurocomputing 159:27–34

Zheng X, Zhang X, Yu Y, Kechadi T, Rong C (2016) ELM-based spammer detection in social networks. J Supercomput 72(8):2991–3005