Turkish Journal of Computer and Mathematics Education *Vol.12 No.3(2021), 4951-4958*

Research Article

Detection of Malicious Data in Twitter Using Machine Learning Approaches

B.Mukunthan[a] , M.Arunkrishna[b,*]

[a]Research Supervisor, Department of Computer Science, Jairams Arts and Science College (Affiliated to Bharathidhasan University), Karur - 639003, Tamilnadu, India.
[b]Research scholar, Department of Computer Science, Jairams Arts and Science College(Affiliated to Bharathidasan University, Tiruchirappalli),Karur - 639003, Tamilnadu, India.
([*]Corresponding author's e-mail: arunkrishna.murugan@gmail.com)

Article History: Received: 10 November 2020; Revised 12 January 2021 Accepted: 27 January 2021; Published online: 5 April 2021

Abstract: Unlike traditional media social media is populated by unknown individuals who can broadcast whatever they like. This online social media culture is dynamic in its nature and transition to digital media is becoming a trend among people. In upcoming years the use of traditional media will decline, and the increasing use of Online Social Networks(OSNs) blur the actual information of the traditional media. The information generated by the authentic users gives useful information to the general users, on the other hand,Spammers spread irrelevant or misleading information that makes social media a plot for false news. So unwanted text or vulnerable links can be distributed to specific users. These false texts are anonymous and sometimes linked with potential URLs. Due to data restrictions and communication categories, the current systems do not deserve an exact statistical classification for a piece of news. We will study different research papers using various techniques for master training in the prediction and detection of malicious data on social networks online. We tried to find spam tweets from the tweets collected by using Enhanced Random forest classifications and NaiveBayes in this research. To evaluate the work, different validation metrics such as F1-scoring, accurcy and precision values are calculated.

Keywords: spammers,Spam detection, Machine Learning , Twitter spam

1. Introduction

Online Social Networks(OSN) has changed the way news is shared between people . This undermines the credibility of the entire news system, because spreading of false information may happen easily. A denotable feature of the OSN is that one can be able to register and publish as many posts as they want even without any upfront cost. Therefore, companies gradually move into social media, and not only users, is best known for celebrities and campaigns which upgrade by enhancing their fans and followers. Fake human and organisational damage profiles decrease their likes and followers count [1]. The specially created software not only posts illegal users, but it also publishes false stories that are spread by followers. False news has a negative impact on the marketing, advertising, and cyber-building benefits of social media. In this era, the social network is one of the most important and local news sources [2]. When people practise spreading false news to make money, however, it is wrong. Almost every country uses the social network as a communication source. False domains and sites are not easy to recognise. Days pass by when bad text is badly sculpted, no padlock symbols, the unlogical URL is the same. Fake resources are currently designed to appear authentic. Regardless, the sources could be a bogus file or a bogus network service. Malicious data is frequently discussed; however, attackers can also trick them into creating data in the absence of individuals. This paper will include several technologies and methods to detect and develop an ML based Twitter Spam Detection Framework for social media malicious data.

Objectives

- To generously analyse various gaps, techniques, and the direction in the field of Spam detection research.
- Offer a method used to determine malicious activity as a combination of ML algorithms
- Twitter Spam Detection Framework design and application
- Improving the strength and efficiency of the framework with pre-processing optimisation function
- Improved robustness through the innovative extraction algorithm.
- Performance assessment by comparison with modern approaches such as F-measurement, accuracy and TPR/FPR Framework

2. Connected Works

[1] has developed a novel method for determining the attitudes of Twitter spam users in order to distinguish spam from non-spam social media posts. By providing independent historical tweets, the proposal made a new contribution. This optimised feature set was offered on Twitter for a limited period. The features were linked to Twitter users, their accounts and arrangements in pairs. The study also determined the strength and efficiency of this optimised set in conjunction with the general functionality of detecting spam. [2] A further range of features has been provided by enhancing the performance of the classificatory in order to identify Twitter spammer. The study analysed the performance of different approaches such as the Support Vector Machine (SVM) in combination with the ML-Tools WEKA and Rapid Miner, K Nearest Neighbor (KNN), and Multilayer Perceptron (MLP). Random Forest (RF). The WEKA offered an impressive response to the Rapid Miner during the experimental set-up. In both cases, the RF classification offered greater performance and was compared to the other.

The Twitter spam detection approach was also designed [3]. The approach is specifically designed to accurately recognise user details. The developed system is also an exclusive feature set and offers increased security controlled by the Google Secure Browsing API. In addition to detect spam, the developed system can also enhance classification performance. The [4] method has also been developed to detect spam in highly accurate URLs using SVM-based Spam maps, spamming images and image spam filters. The research also proposed that a phishing system based on URl should use popularity at the site, host-based features and verbal abilities. Different prominent algorithms have already been used such as the Artichipelago, the Decision Tree, the SVM, the KNN Classifier, the Random Forest, and Logistic Regression. A number of key algorithms are also used.

Machine learning algorithms have been proposed for spam identification is based on Latent Dirichlet Allocation concept [5]. The system has been manually tested with 15,000 spam words and 6320 non-spam words. The uniqueness of the tweets is that they can be easily tracked using machine learning algorithms, which is seen as a system benefit. In addition, numerous methods for spam detection were used and high precision and increased detection rates tested and verified.

[6] A spam detection system, known to Oasis using an online scalable method, was developed for social network spam. Due to the two key compounds, innovative systems have been developed. A decentralised DHT-based tree overlay was used to detect and collect spam in social communities as the source. The latter effectively distinguishes new spam and combines spam message properties to produce spam classification devices. In this study, the Oases model was developed and then used. Tweeters are experimentally configured with large realm data. The results are superior in terms of efficiency, eye-catching load balancing and scalability in social network online spam detection. [7] for identification of the Twitter spammer was proposed a unique semi-supervised framework, called Spam2Vec. The algorithm framework by utilising prejudicial random pathways, which can identify the spam image in the network node. This technique has been shown relatively better compared with other primary and benchmark approaches.

[8] The model proposed a Twitter drifted spam model. This model can detect spam over time. spam can be detected. Drift detection (MDD) and divergence of the KL have been used to identify the drifts. Moreover, accuracy, reminder and the f-measurement have been found with higher precision while maintaining the base class with detected results. An INB-DEN-Stream (Den-Stream and Incremental Naive Bayes.) hybrid combination is used for [9]. It uses stream clusters to group spam and ham tweets. The clusters in this method are referred to as symmetrical or asymmetrical micro clusters. In addition the Euclidean distance was replaced by a number of classifiers. And assess the performance using the denstream and StreamKM++ classifiers, to show comparison.

[10,11] A semi-controlled learning method for spam detection has also been introduced. In this approach, the framework was stopped by probabilistic data structures (PDS). In addition to the Quotient Filter (QF), Locality Sensitive Hashing (LSH), which was also used for spam word database and URL queries. The LSH was used for the assessment of similarity. In addition, parameters such as F-score, precision and recall, have demonstrated that the model is more successful. The proposed framework also restricts the calculation process. In a similar field of research, [12] proposed another hybrid system. This system uses the set of functions to use and analyse important information. The Google Safe Browsing API is used for increased security. New features in which the SVM amalgamation with NB naive Bayes was mixed with twitter4j are used for the desired solution.

3. Methodology

Exploratory Data Analysis (EDA)

The EDA tool was used to carry out the primary data survey, an important approach. EDA requires a number of techniques and approaches for data analysis. This approach enables you to construct the patterns, eliminate the abnormalities and make a hypothesis. The results are shown by graphical approach and statistics have been used to develop the assumption. To sum up, the EDA consists of compiling the statistics and raw information and

maximising the ability to recognise patterns through plot positions. This is an essential process in terms of business understanding and connectedness. The useful steps for open-source Word cloud are also included in this section.

Getting tweets from data sets

The records in the data set are instances of certain data that are organised in a particular way. The data set entries are linked to a specific type of information and are recognised with certain types of data structure. On Twitter, spam and ham, there are two different types of spam detector data sets. There are sometimes problems with authenticity prediction in ham messages, these messages are usually called harmful (i.e. hard to know them). This is therefore incorrectly classified as spam in several cases. Please note that the data set Kaggle, AWS data package Kaggle and UC Irvine are free of charge.

Exploration and Analysing the Data

For a data analyst in research, this is a key step in understanding and exploring what is happening in the data set. The main characteristics of any data can be examined at this stage. Therefore, it is vital to explore meaningful patterns and features for large and dispersed datasets.

Visualization

Diagraphs, diagrams and plots can be understood visually through this process. Visualization therefore makes it easy to understand. Furthermore, in Twitter, the N-gram is one of the prominent examples which helps to identify the magnitude of words that are considered to be one unit to better develop model predictions with the highest accurately. It subsequently divides datasets in two more units, i.e. I sets of training and ii) tests. You can also simplify the visualisation process via a word cloud.

Data Pre-processing

Pre-processing data includes transforming raw data into a more explanatory form in the field of machine learning. The pre-processing helps to eradicate data noise, since the mix is made reliable by mixed with the real data. The process involves text insertion, selection of functions and normalisation of the cleaning process. In addition, preprocessing can achieve better results in ml models. Different data set noise can be removed via Text Cleaning, such as hyperlinks, whitespace, punctuation and numbers. The standard processes here include conversion of lowercases, eradication of white and dotted spaces and numbers. In addition, there are also word lemmatization and word streaming. Standardization is essentially a process in which text documents are prepared for NLP events. Two major normalisation methods, such as lemmatization and stemming, can be used to identify the word root forms.

Word lemmatization

This minimises inflection, also by means of morphology and vocabulary analysis of these words the basic(rooted) form of the word is calculated. This is done by using a specific language dictionary and the words are converted in their primary format. Substantial thinking and pre-planning are an obstacle to the appropriate application of these algorithms. However, the process can be carried out with the help of NLTK library easily.

Word stemming

Stemming is the process of reducing a word to their base form also known as written form. The stem is the morphological core of the word called as lemma. in order to map a word it is necessary to cut off it’s suffix or prefix. This heuristic technique is expected not to be always perfect as some tomes over-fitting or under-fitting also happen.

Extraction of Features

It is critical to get text data ready for machine learning. To eliminate words from text data, special consideration is required. The word Tokenization is referred to for such a technique. The text must be transposed into numbers because we cannot handle the text directly in machine learning. This is why science-study tokenization and additional functional extraction clears. A remarkable Bag of Words (BoW-ECM) technology was based in this regard on the number of word incident. Algorithms typically accept numeric values (int or float), so extraction layers convert words to "int." This is accomplished through the use of popular methods such as words embedding, Tfdifvectoriser, and countvectorizer.

Algorithm implementation and Training

Numerous algorithms flood the literature and we must examine their description thoroughly. The choice of a more appropriate model can halve the work. The language of implementation is very important and needs to be

carefully selected, because APIs and libraries depend immediately on their application. The implementation language is very important. Training provides the model with the strength to predict correctly. The Deep Learning algorithm uses a dataset of training. Based on an educated model, new insights can be created and predicted. Three types of training are specifically classified for machine learning models. Patterns are observed during the training to allow the system to predict new data through the targeted attributes. In Machine Learning, optimal training is needed to produce effective classification results.

Naive Bayes Algorithm

The Bayesian theorem is implied in such an algorithm in order to classify objects. Various other algorithm types are also based on the same principle. Simple/Independency Bayes is also known as this type of classifier. Such a classifier is a classic ML-based technology and can be commonly used for spam filtering. This classifier can be used between attributes of data points to achieve the strong or weak (naive) independence. This has several objectives, such as medical diagnosis, text classification and sample detection. The Naïve Bayes classification is not one of the widest methods based on machine learning. This is based on the principle of basic probability and the Bayes model is used to govern later. Spam detection and text grading are the reason for its popularity.

The framework can compute the probability in each class and then most probably return the class. This happens through features like$(X_0, X_1, \ldots, X_m)$, and the classes like$(C_0, C_1, \ldots C_n)$. The distribution of probability will therefore be $P(C_i \vee X_0, X_1, \ldots, X_m)$for each class. The Bayesian rule is as follows

$$P(A \vee B) = \frac{P(B \vee A)P(A)}{P(B)} \qquad (1)$$

Here, in Equation1, 'A' denotes the classes and 'B' denotes features. The say class $(C, C_1, \ldots, C_n)$can be replaced by a feature $(X_0, X_1, \ldots, X_m)$.The standard $P(B)$ is very difficult to compute here, so instead we will take into account,

$$P(C_1 \vee X_1, X_2, \ldots, X_m) \propto P(X_1, X_2, \ldots, X_m) * P(C_i) \qquad (2)$$

Here, $P(C_i)$ portion represents a dataset which is easy to calculate. In contrast, it's rather hard to calculate $(X_0, X_1, \ldots, X_m \vee C_i)$. In order to make its calculation easier, let us presume that we have $(X_0, X_1, \ldots, X_m)$and that these depend on the C_i case, So, it is possible to mention;

$$P(X_0, X_1, \ldots, X_m \vee C_i) = P(X_0 \vee C_i) * P(X_1 \vee C_i) \ldots P(X_m \vee C_i) \qquad (3)$$

This may not always be the case and is therefore labelled as Naive Bayes classification. Below is the probability of class.

$$P(C_i \vee xX_o, X_1, A_1) \propto P(X_0, X_1, \ldots, X_m \vee C_i) * P(C_i) \qquad (4)$$

i.e. $P(C_1 \vee X_1, X_2, \ldots, X_m) \propto P(C_i) \prod_{j=1}^{m} P\left(X_j \vee C_i\right)$

We have determined calculation for $P\left(X_j \vee C_i\right)$. That follows the multinomial distribution, as the word count classification is treated as word count. In case of continuous functionality, it will have Gaussian distribution. In comparison to other algorithms, NaiveBayes algorithm doesn't need explicit training. This algorithm is also suitable for the handling of large data points and high-dimensional data points.

Classification Method

This algorithm also makes the categorization very effective and easy. The probability of certain data points can simply be estimated and can also easily be compared with the classes. Fix the Ci, according to the higher chance,

$$y = argmax P(C_i) \prod_{j=1}^{m} P\left(x_j \vee C_i\right) \qquad (5)$$

The maximum post-decision rule is known as Eq. 5.

Posterior Probability: The Bayesian post-probability statistics are considered as a conditional probability for a random variable. This is usually calculated by finding and considering background evidence. Maximum posterior probability: this calculates the quality of the unknown, which later is equivalent to the model of probability after distribution.

ERF (Enhanced Random Forest) Algorithm

This algorithm is monitored using the fundamental decision-tabling principle. The decision tree is therefore considered to be the building blocks for this classificatory. Usually this method is used when class information is not provided. This can be used as a known method for preventing classification and regression problems; for this purpose it is helpful to create a decision tree. The accuracy of this algorithm is connected linearly by its dimensions, which increase when a decision-making table is increased. This works with a set of rules to support

the IG and Gini Index process (GI). Here, unlike the data, the entropy, called randomness, is measured. In this case, if the splits are bigger, the prediction is better. You can measure this entropy.

$$H = -\sum p(\quad) log p(x) \qquad (6)$$

Here in Equation 6, H is the Entropy,$P(x)$shows the percentage of a particular class in a specific group. The more classes the more entropy, or if the case is one class the more entropy the more the other. Entropy also plays a key role in the collection of data. The IG is a measure which shows the amount of information that can be stored for a specific class within a feature.

$$Gain(S, D) = H(S) - \sum_{V \in D} \frac{(V)}{(S)} H(V) \qquad (7)$$

Here, S is current set Split is denoted by D and V denotes the subset of S. Information Gain can identify best division here. In addition, GI is an impurity measure. The Gini is based on the decision tree level. The leaf level is low. The impurity of Gini node is,

$$I_G(n) = 1 - \sum_{i=1}^{J} (P_i)^2 \quad (8)$$

In Equation 8, we can obviously use the J and Pi values to calculate the impurity of the gini. The CART and the RF are ideal for improving non-linear connections. The classification problems may be further determined and the regression calculated. Because the CART is highly sensitive to the chosen variable and predictors.

Algorithm

Step No. 1: Initialization of Parameter
Folds:
Seeds:
Number of trees:
Maximum depth of tree:
Br:
Step No. 2 : Import the data set into buffer i.e. Br.
Step No. 3: Buffer reader, Br = null;
Step No. 4 : Br = buffer reader(file reader);
Step No. 5: Extract the features.
Step No. 6: Set number of trees = 10;
Step No. 7: Number of folds = 10;
Step No. 8: Set max depth = 0
Step No. 9: Implementation of cross-validate model (data) for evaluation of results
Step No. 10: For the evaluation of results, calculate TP, F-Measure, FP
Step No. 11: Distinguish between Ham and Spam
Step No. 12: End

SystemDesign

A Framework for out proposed Twitter Spam Detection System is shown in Figure1. In this context, the optimised characteristics for pre-processing are considered with reference feature optimization. These features can only be presented on Twitter for a short period of time. These features deal with the pair arrangement of the respective accounts and users. Whilst this was a better system for strength and efficiency compared with several other features of spam detection. To extract these characteristics, CountVectorizer is used. In several other neural networks, this algorithm is regarded as the priority due to its higher learning barriers handling ability. Later on, ERF Classifier will group the features. This enables real numbers to be trained and testing tasks to be done because it parallels resources. RF is also much better than other rating systems.

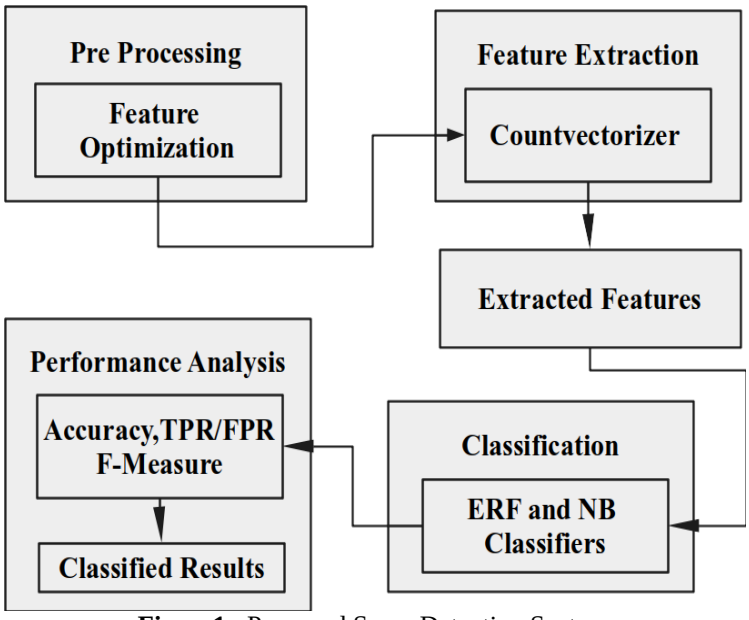

Figure1 : Proposed Spam Detection System

4. Scoring and Metrics

After successful completion of the training, the model can be confirmed. The assessment process helps us to assess the performance of the model on untrained data. The dataset we hold off often plays beforehand. The evaluation metric assists us in calculating the model performance into real data. Only the accuracy measure is not considered a better assessment. For example, of 100, only twenty are spam, even if the algorithm does not acknowledge any of them as spam the accuracy is 80. Similarly, if the data set only includes one out of 100 spam, then 99 percent accuracy will be achieved by the algorithm considered as non-Spam. Therefore, precision is not worth measuring the performance based upon accuracy, and reminder is used as performance indicators in this case.

Precision and Recall

This measures how correct the positive identifying ratio is. Precision can be calculated with the help of Eq. 9.

$$\text{Precision}=\frac{\text{TruePossitive}}{\text{TruePossitive}+\text{FalsePositive}} \quad (9)$$

Recall, often called as "True Positive Rate." is the ratio of the correct and wrong prediction. is as a coefficient. The recall can be calculated by Eq.10.

$$\text{Recall}=\frac{\text{TruePossitive}}{\text{TruePossitive}+\text{FalseNegative}} \quad (10)$$

ConfusionMatrix

Forecasted findings can be summarized using a confusion or error matrix matrix. The performance is measured using some essential metrics, such as retrieval, measurement, prediction and precision. The matrix for confusion can be used with the Eq.11.

$$\text{Accuracy}=\frac{\text{TruePossitive}+\text{TrueNegative}}{\text{TruePositive}+\text{TrueNegative}+\text{FalsePositive}+\text{FalseNegative}} \quad (11)$$

It helps to understand results and to contemplate performance of the model and to capture the results. Informative methods and techniques of plotting are provided by Scikit learned.

$$F-\text{Measure}=\frac{2\times\text{Recall}\times\text{Precision}}{\text{Recall}+\text{Precision}} \quad (12)$$

Eq. 12 shows, for precision and reminder, an average called weighted harmonic means.

5. Results and Discussion

To gather data for this study, various performance measures such as accuracy, accuracy, and f-measurement were used. The results were compared to 20 to 40 percent spam messages in this study. The results show that this model was intuitively well-executed and has a classification precision of more than 90%. Our proposed model has been established to provide the best values in accuracy score. Our model achieved a precision of over 95 per cent with optimum training. In similar ways, NaiveBayes' accuracy and recall values are 45%, but 90% with ERF. In

addition to this, with carefully tuned training,F-Measure of NB is around 60% and for EnhancedRF it's score is as high as of 85% were achieved. The results shown in Figure 2 and Table 1 shows that the proposed technique is efficient in terms of detecting spam messages.

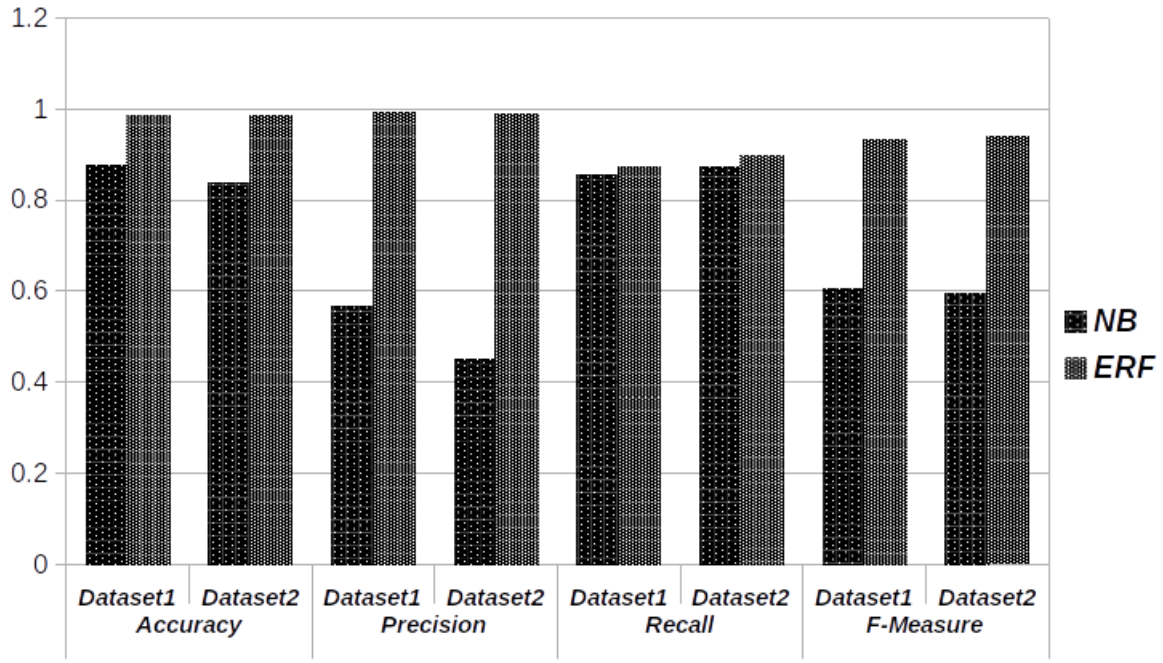

Figure 2: Performance Metrics

Table 1: Analysis of Results

Metrics	Datasets	NB	ERF
Accuracy	Dataset_1	0.8745519713261649	0.982078853046595
	Dataset_2	0.8351254480286738	0.9838709677419355
Precision	Dataset_1	0.5649122807017544	0.98967860342758
	Dataset_2	0.44966442953020136	0.9857142857142858
Recall	Dataset_1	0.8548387096774194	0.8709677419354839
	Dataset_2	0.8701298701298701	0.8961038961038961
F-Measure	Dataset_1	0.6022727272727273	0.9310344827586207
	Dataset_2	0.5929203539823009	0.9387755102040817

6. Conclusion

In the past, the systems were focused mainly on identifying spammers in the profiles of Twitter users and social honeypots. which focused on users' accounts, historical tweets and social diagrams. With the current technologies, spammers can easily invade in to OSNs and spread malicious contents. So that we need some complex methods Such as ML and deep learning techniques to resist spams. As we discussed earlier ,malicious data will posses more threat to the online social network users so this paper proposed Machine Learning based spam detection system,that can capable of effectively filter twitter spam. In this way the study concludes that machine learning techniques provide better results in the detection of malicious messages in social media compared with profound learning. In our future work, we will incorporate some hybrid ML classification to prevent spam

References

J.C.S. Reis, A. Correia, F. Murai, A. Veloso, F. Benevenuto, E. Cambria, Supervised Learning for Fake News Detection, IEEE Intell. Syst. 34 (2019) 76–81. doi:10.1109/MIS.2019.2899143.

D. Ramalingam, V. Chinnaiah, Fake profile detection techniques in large-scale online social networks: A comprehensive review, Comput. Electr. Eng. 65 (2018) 165–177.

S. Paul, J.I. Joy, S. Sarker, A.A.H. Shakib, S. Ahmed, A.K. Das, Fake News Detection in Social Media using Blockchain, 2019 7th Int. Conf. Smart Comput. Commun. ICSCC 2019. (2019) 3–7. doi:10.1109/ICSCC.2019.8843597.

M. Sowmya, J. Shiva Shankar, A Survey on Detection of Fake News in Social Media, Int. J. Res. Available. 6 (2019) 469–474. https://journals.pen2print.org/index.php/ijr/ (accessed October 1, 2020).

S. Rauti, V. Leppanen, A Survey on Fake Entities as a Method to Detect and Monitor Malicious Activity, Proc. - 2017 25th Euromicro Int. Conf. Parallel, Distrib. Network-Based Process. PDP 2017. (2017) 386–390. doi:10.1109/PDP.2017.34.

X. Zhou, R. Zafarani, A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities, ACM Comput. Surv. (2020). doi:10.1145/3395046.

K. Shu, A. Sliva, S. Wang, J. Tang, H. Liu, Fake News Detection on Social Media, ACM SIGKDD Explor. Newsl. 19 (2017) 22–36. doi:10.1145/3137597.3137600.

W.Y. Wang, "Liar, liar pants on fire": A new benchmark dataset for fake news detection, ACL 2017 - 55th Annu. Meet. Assoc. Comput. Linguist. Proc. Conf. (Long Pap. 2 (2017) 422–426. doi:10.18653/v1/P17-2067.

N.J. Conroy, V.L. Rubin, Y. Chen, Automatic deception detection: Methods for finding fake news, Proc. Assoc. Inf. Sci. Technol. 52 (2015) 1–4. doi:10.1002/pra2.2015.145052010082.

A. Varol, O., Davis, C. A., Menczer, F., & Flammini, Feature engineering for social bot detection. Feature engineering for machine learning and data analytics, 311th ed., 2018.

C.M.M. Kotteti, X. Dong, N. Li, L. Qian, Fake news detection enhancement with data imputation, Proc. - IEEE 16th Int. Conf. Dependable, Auton. Secur. Comput. IEEE 16th Int. Conf. Pervasive Intell. Comput. IEEE 4th Int. Conf. Big Data Intell. Comput. IEEE 3. (2018) 193–199. doi:10.1109/DASC/PiCom/DataCom/CyberSciTec.2018.00042.

A. Bondielli, F. Marcelloni, A survey on fake news and rumour detection techniques, Inf. Sci. (Ny). 497 (2019) 38–55. doi:10.1016/j.ins.2019.05.035