International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

An Effective Twitter Spam Detection Model using Multiple Hidden Layers Extreme Learning Machine

[1*]Dr. K. Maithili, [2]T. Prabhakara Rao, [3]C. Ambhika, [4]Y. Divya, [5]Bommisetti.Yamini Supriya, [6]Dr. R. Sundar, [7]Tabish Rao, [8]Dr. Balajee J.

Submitted: 20/06/2023 **Revised:** 02/08/2023 **Accepted:**25/08/2023

Abstract: In contemporary times, social networking sites have gained widespread popularity as tools for interaction and communication. Among these platforms, Twitter holds a significant position, facilitating news consumption, idea sharing, social discourse, and interpersonal communication. However, due to its wide user base, Twitter has also become a breeding ground for spam activities. Numerous studies have been conducted to detect spam on Twitter, employing both traditional and machine learning models. Addressing this issue, this paper introduces an innovative approach to Twitter spam detection using a multi-layered extreme learning machine (MELM). Additionally, the Word2Vec model is employed to map words in the dataset into multi-dimensional vectors. By introducing multiple hidden layers and adaptively initializing weights connecting input, first hidden layer, and bias, the MELM model advances beyond the conventional ELM model. The application of the least squares technique aids in determining output weights for the network. To assess the efficacy of the MELM model in detecting spam, extensive experiments were conducted on three spam datasets. The results demonstrate the MELM model's proficiency, achieving an accuracy of 0.8817, precision of 0.9057, recall of 0.8650, and an F-Score of 0.8848.

Keywords: *Twitter, spam detection, machine learning, word2vector, extreme learning machine*

1. Introduction

People have been signing up for online social networks (OSNs) more frequently lately. These websites, including Facebook, Twitter, and Instagram, have changed how people engage with one another and communicate with one another. Additionally, many industries have utilized OSNs as an ad and promotion instrument to increase selling. Twitter is famous widespread OSNs with incredible popularity having up to 313 million active clients based on the newest data. At the same time, the huge development of Twitter permits growing number of clients to share their data and connect together. Yet, the allure of Twitter's popularity has attracted spammers, resulting in the proliferation of such malicious actors. These Twitter spammers typically refer to users who post tweets containing advertisements, illicit drug sales, or messages that direct users to malicious external links [1]. This might cause phishing attacks or malicious uploads, etc.

Spammers on Twitter are not affecting the online social practice, as well threatened the privacy of cyber space. For instance, in September 2014, New Zealand's network system is melted that caused malicious uploading spam, the outcome that signaled the warning of Twitter spammers. Later, the enormous amount and higher threats of Twitter spam are crucial to be prohibited. To efficiently reduce spam activity in Twitter, several Twitter spam identification techniques are introduced, involving 1 with Twitter itself. To better skill, there are 3 significant types of approaches for Twitter spammer identification. In initial category, technique depends on client accounts and tweet content features. This feature is simply removed from tweets by little or no calculation. However, this feature is simply invented. The next category depends on strength feature resulting from the social graph that goals to discover the connection of receivers and senders. Hence, graph-based feature is analytically tedious to gather since creating a huge social or relationship graph is time and asset consumption. This is because of a mistake that a client might communicate by large but impulsive number of clients. The 3rd group emphases on tweets by URLs. For instance, IP blacklists and domains are extensively

[1*]*Associate Professor,Department of CSE,KG REDDY COLLEGE OF ENGINEERING &TECHNOLOGY,Moinabad, Hyderabad, Telangana-501504. drmaithili@kgr.ac.in.*
[2]*Associate Professor,Department of Computer Science and Engineering,Aditya Engineering College (A),Surampalem, Andhra Pradesh,533437. prabhakar.tatapudi@gmail.com.*
[3]*Assistant Professor,Department of Information Technology, Velammal Institute of Technology,Ponneri, Chennai. ambhidurai@gmail.com.*
[4]*Sr. Assistant Professor,Department of EIE,CVR COLLEGE OF ENGINEERING, HYDERABAD. y.divya@cvr.ac.in.*
[5]*Assistant Professor, Department of CSE,Koneru Lakshmaiah Education Foundation,Vaddeswaram,AP,India. byamini@kluniversity.in.*
[6]*Assistant Professor,Computer science and Engineering,Madanapalle Institute of Technology & Science,Ap. drsundarr@mits.ac.in.*
[7]*Assistant Professor, School of Computing, Graphic Era Hill University, Dehradun, R/S, Graphic Era Deemed To Be University, Dehradun, 248002. tabishrao0609@gmail.com.*
[8]*Associate professor,Department of Computer Science,Mother Theresa Institute of Engineering and Technology, Palamaner,Andhra Pradesh. balajeej04@gmail.com.*
Corresponding Author's Mail:

utilized to filter tweets involving malevolent URLs. But most current scientists highlighted theoretic research. The absence of appropriate Twitter spammer identification techniques that allow effective online identification to fight spammer in real-time.

Several investigations have employed both conventional and machine learning (ML) approaches to detect spam on Twitter. To contribute to this field, our paper introduces a novel method for identifying Twitter spam, utilizing a multi-layered extreme learning machine (MELM). Furthermore, we leverage the Word2Vec model to convert the words within the entire dataset into corresponding multi-dimensional vectors. The MELM model expands upon the original ELM model by incorporating additional hidden layers, initializing input-to-hidden, first hidden layer, and bias weights in a flexible manner. It employs the least squares technique to calculate the output weights of the network. To gauge the enhanced efficacy of the MELM model in spam detection, an extensive array of experiments was conducted on three distinct spam datasets.

2. Literature Review

In previous research, [2] introduced four machine learning (ML) classification methods, along with various general account and text-dependent attributes, to discern spammers on Twitter. Additionally, [3] discussed four ML classifier techniques combined with thirteen text-dependent features. Similarly, [4] devised two approaches to identify spam profiles and tweets. In their initial phase, they formulated a system to identify spam profiles based on account-related attributes. Through the application of Support Vector Machine (SVM) classification, their model achieved an accuracy rate of 84.5%. Subsequently, they extended this framework to incorporate both account-based and text-dependent features for classifying tweets into spammer and non-spammer categories.

In another study, [5] introduced ten machine learning (ML) classification methods along with two separate datasets for identifying spam. Furthermore, [6] collected a massive dataset comprising more than 600 million public tweets. They then derived twelve lightweight features from this dataset and employed six ML techniques to categorize tweets into spammer and non-spammer classifications. Also, [7] proposed 6 light weight features and distinct ML techniques. While these techniques denote better advantages simple removal of tweet text and account dependent feature, it every suffer from similar disadvantages. In fact, spammers can prevent this feature to avoid identified. To resolve this issue, scientists have leverage on the social graph to arise with high strength. [8] proposed following and follower relations. They removed near 25K Twitter clients, and 20 new tweets of every client, together 49M friends or follower's relations. To evaluate recognition technique, they utilize 4 distinct classifications. In the researches, NB classifiers provide the optimum efficiency of 91.7% accuracy and 91.7% f1-score. [9] introduced a hybrid architecture, utilizing clients meta data, additionally to graph and tweet text depend feature to identify spammer on Twitter. They utilized 19 features and three common ML techniques such as RF, DT, NB to classify clients into spammer and non-spammer.

In a separate study, [10] introduced four categories of characteristics including account, text, graph, and automated-based features (for instance, utilizing frequency from Twitter API), and employed three ML classifier techniques to differentiate between spammer and non-spammer users. Their research resulted in an impressive 90% f1-score. Another approach by [11] incorporated the Word2Vec deep learning method and an additional Doc2Vec training technique to assign a multi-dimensional vector to each tweet. [12] innovatively developed an attribute engineering system using a deep neural network (DNN) with a Bi-directional Long Short-Term Memory (Bi-LSTM) technique. They trained the DNN with labeled datasets and extracted features from hidden layers to represent tweets. The outcomes were compared with other methods, demonstrating optimal results. Correspondingly, [13] proposed a deep learning-based function that involves concatenating word vectors in a convolution layer. In a similar vein, [14] utilized convolution techniques on sentences with hierarchical architectures. [15] introduced modifications to the LSTM architecture, designing a tree-structured topology by stacking CNN and LSTM layers sequentially, resulting in effective outcomes for semantic sentence modeling.

3. The Proposed Spam Detection Dataset

The presented model primarily receives the raw Twitter data as input and performs preprocessing to improve the data quality. Next, Word2Vec model is applied for mapping the words in the entire dataset into respective multi-dimensional vectors. At last, MELM model gets executed to determine the existence of spammers or non-spammers in the Twitter dataset. Distinct from conservative spam detection process, the features extraction process can take place based on the content of twitter utilizing Word2Vec in the place of classical characteristics collection and generation. Primarily, employ Word2Vec to map every word in the entire datasets with equivalent multi-dimensional vectors. It utilizes 2 level neural networks and Huffman method utilizes hierarchical-SoftMax for allotting codes to the common words. It enhances the effective training method; as maximum frequency word might process rapidly. By employing this method, the word vector dependent indications are trained by stochastic gradient descent, and

it is attained via backpropagation (BP). The optimum vector is attained to every word through Skip-gram /CBOW. Besides, Doc2Vec training technique is utilized to allocate 1 vector denoting that all the tweets have utilized Paragraph Vector model.

As per the Word2Vec approach, a document vector for tweets of varying lengths is trained using a combination of words, generating distinct document vectors for each report. Through iterative processes, the most effective document-based vectors are identified. Subsequently, the document vectors are refined through high-level dimension learning, serving as input attributes for various machine learning methods such as Neural Networks (NN) and Random Forests (RF), along with their corresponding spam or non-spam labels. This denoted document vector is represented as "d"

$$D=d1, d2, \ldots,dM, \quad (1)$$

Here, "M" signifies the dimensionality of the document vector, and "d" represents the values associated with each level. Moreover, when incorporating binary labels as an additional parameter, the tweets are characterized as follows

$$t=D, label, \quad (2)$$

where t indicates concatenate vector and tweet flag of spam or non-spam label. Therefore, T the training datasets are represented as

$$T=t1,t2, \ldots,tN, \quad (3)$$

In this context, "N" represents the total count of tweets within the training datasets. Leveraging these training datasets, a binary classification function labeled as "C" is constructed through conventional machine learning methods. The purpose of this function is to forecast features for unlabeled testing data. This prediction is facilitated by utilizing a label vector denoted as "L," organized in alignment with the sequence of the respective messages. This process is succinctly represented as "b."

$$L=l1, l2, \ldots, ln=CD1, D2, \ldots, Dn, \quad (4)$$

where n denotes tweet number in data testing.

Recently, researchers have made significant advancements in ELM techniques and architectures, leading to notable achievements. These accomplishments have inspired a closer examination of the most effective concepts within the ELM framework. In light of this, a new technique named MELM has been introduced. The architecture of the MELM, which involves a 3-hidden layer ELM, is illustrated in Figure 1.

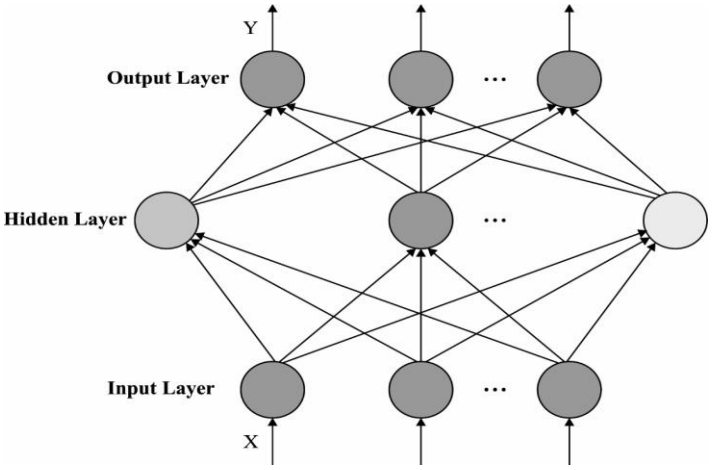

Fig. 1. Structure of MELM model

Consider a 3-hidden layer Extreme Learning Machine (ELM) and explore its behavior within the Multi-layer ELM (MELM) technique. The training samples, designated as "X," and the corresponding targets "T = xt, tii = 1,2,3, …,Q" are provided. This architecture comprises a 3-hidden layer network with "l" hidden neurons per layer and the activation function "gx." Comprising input, output, and 3 hidden layers, it serves as the foundation. To implement the Transfer ELM (TELM) technique's concept, the 3-hidden layers are reconfigured into 2-hidden layers. The initial hidden layer remains unchanged, while the 2nd and 3rd hidden layers are merged into a single hidden layer. The primary objective is to establish the weights matrix "βnew" that connects the 2nd hidden layer to the output layer. Through considering the original sample count, weights "β" are estimated. This approach enhances the network's ability to simplify tasks, ultimately leading to improved performance.

Subsequently, the MELM technique divides the previously combined 3-hidden layers, resulting in a MELM architecture with three distinct hidden layers. As a result, the anticipated output of the 3rd hidden layer is depicted as:

H3=Tβnew+. (5)

The term "βnew+" represents the general inverse of the weight matrix "βnew." Subsequently, within the MELM framework, the matrix "WHE1" is computed as the product of "B2" and "W2." This results in the variables of the 3rd hidden layer being straightforwardly obtained by estimating Equation (5). This can be expressed as: "H3 = g(H2W2 + B2) = g(WHE1HE1).

WHE1=g-1H3HE1+, (6)

In this context, "H2" denotes the initial output of the 2nd hidden layer, "W2" represents the weight linking the 2nd hidden layer to the 3rd hidden layer, and "B2" indicates the bias of the 3rd hidden neurons. "HE1+" corresponds to the general inverse of "HE1 = 1 H2T," where "1" signifies a one-column vector of size "Q," with components being scalar units of "1." Moreover, "g^-1(x)" denotes the inverse of the activation function "gx." To assess the efficacy of the introduced MELM technique, diverse activation functions are applied for regression and classification tasks, as described in [16]. Particularly, the logistic sigmoid function gx = 1 / (1 + e^-x) is commonly utilized. The initial output of the 3rd hidden layer is calculated as follows:

H4=gWHE1HE1. (7)

Lastly, the output weight matrix βnew among 3rd hidden and output layer is estimated by while the count of hidden layer neurons lesser when compared to number of training sample, β which is represented by:

βnew=Iλ+H4TH4-1H4TT. (8)

Although count of hidden layer neurons higher over the count of training sample, β is denoted by:

βnew=H4TIλ+H4H4T-1T. (9)

The original output of 3 hidden layer ELM network is given by:

fx=H4βnew. (10)

To ensure that the initial output of the last hidden layer accurately predicts the hidden output in the training model, an optimization procedure is initiated with the network architecture variables, commencing from the 2nd hidden layer. This process encompasses estimating the variables of the 3-hidden layer ELM network and subsequently determining the variables of the MELM network, including its final output. To illustrate this estimation process within the context of MELM, a cyclical estimation concept is employed. When a 4-hidden layer ELM network is introduced, the computations from Equation (5) to (9) are recalculated as part of the network's estimation procedure. This process is reiterated as the number of hidden layers increases, ensuring consistent and coherent estimation operations.

Algorithm 1: MELM algorithm

(1) Let's consider a training sample dataset denoted as "X" with corresponding targets "T = xt, tii = 1,2,3, …,Q." If we represent the input samples as matrix "X" and the labeled samples as matrix "T," each hidden layer in the architecture comprises "l" hidden neurons activated by the function "gx".

(2) Start the process by irrationally allocating the weights (denoted as "W" between the input layer and the first hidden layer) and the bias (denoted as "B" for the first hidden neurons). The formulas for this are "WIE = B" and "W,XE = 1 XT."

(3) Estimate the equation H=gWIEXE.

(4) Estimate the weight among hidden and output layer β=I/λ+HTH-1HTT or β=HT(I/λ+HHT)-1T.

(5) Estimate the predictable output of 2nd hidden layer H1=Tβ+.

(6) Estimate the weight W1 between the first and second hidden layers and the bias B1 of the second hidden neurons using the algorithm steps (4, 5).

(7) Attain and upgrade the original output of 2nd hidden layer H2=gWHEHE .

(8) Upgrade the weight matrix β among hidden and output layer βnew=I/λ+H2TH2-1H2TT or βnew=H2TI/λ+H2H2T-1T.

(9) if the number of hidden layers is 3, estimate the variables via reprocessing execution of the above process from step (5) to (9). Now βnew is given as βnew=β, HE=1 H2T.

(10) Estimate the output, fx=H2βnew.

4. Performance Validation

The efficacy of the MELM model is evaluated using three distinct spam datasets. The features incorporated in the dataset are outlined in Table 1. Further information regarding these three datasets is presented in Table 2. Specifically, Dataset 1 encompasses 1000 spam and 1000 non-spam tweets, Dataset 2 involves 10000 spam and 10000 non-spam tweets, and Dataset 3 encompasses a total of 100000 spam and 100000 non-spam tweets.

Table 1 Attributes description

Feature name	**Description**
Account-based features	
account_age	The age of an account
no_follower	# of followers
no_following	# of followings
no_userfavorites	# of favorites the user received
no_lists	# of lists in which the user is a member of
no_tweets	# of tweets that has been posted by the user
Content-based features	
no_ retweets	# of times this tweet has been retweeted
no_tweetfavorites	# of favorites this tweet received
no_hashtag	# of hashtags in this tweet
no_usermention	# of times this tweet being mentioned
no_urls	# of URLs contained in this tweet
no_char	# of characters in this tweet
no_digits	# of digits in this tweet

Table 2 Dataset description

Dataset	**Spam Tweets**	**Non-spam Tweets**
1	1000	1000
2	10000	10000
3	100000	100000

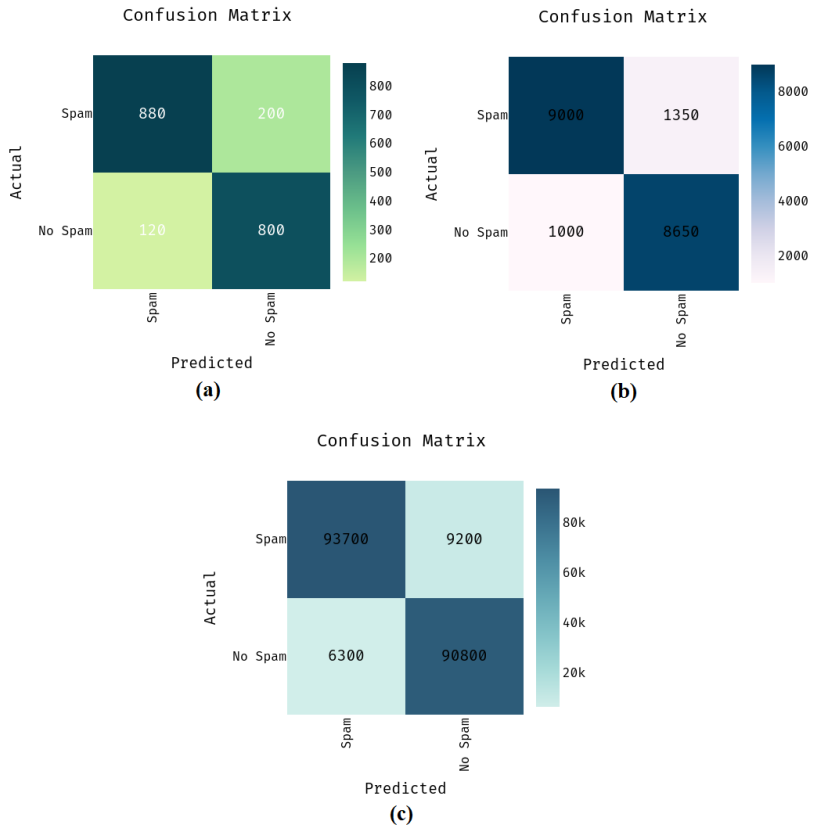

Fig. 2. Confusion Matrix a) Dataset-1 b) Dataset-2 c) Dataset-3

Fig. 2 shows the confusion matrices generated by the MELM model on the applied three datasets. Fig. 2a illustrates that the MELM model has proficiently detected 880 tweets as spam and 800 tweets as non-spam. Besides, Fig. 2b showcases that the MELM method has proficiently detected 9000 tweets as spam and 8650 tweets as non-spam. Moreover, Fig. 2c demonstrates that the MELM method has proficiently detected 93700 tweets as spam and 90800 tweets as non-spam.

A succinct analysis of the detection results achieved by the MELM model on the applied dataset is presented in Table 3. The MELM model demonstrated effective identification of both spam and non-spam tweets within the dataset. For instance, on the test dataset-1, the MELM model exhibited strong detection performance, achieving a maximum accuracy of 0.8400, precision of 0.8800, recall of 0.8148, and F-score of 0.8462. Similarly, on the test dataset-2, the MELM technique showcased notable detection outcomes, achieving higher accuracy of 0.8825, precision of 0.9000, recall of 0.8696, and F-score of 0.8845. Additionally, on the test dataset-3, the MELM method demonstrated effective detection performance with superior accuracy of 0.9225, precision of 0.9370, recall of 0.9106, and F-score of 0.9236. Notably, the MELM model achieved an overall average accuracy of 0.8817, precision of 0.9057, recall of 0.8650, and F-score of 0.8848.

Table 3 Analysis of the Proposed Method's Results for the Applied Dataset

Metrics	**Dataset_1**	**Dataset_2**	**Dataset_3**	**Average**
Accuracy	0.8400	0.8825	0.9225	**0.8817**
Precision	0.8800	0.9000	0.9370	**0.9057**
Recall	0.8148	0.8696	0.9106	**0.8650**
F-Score	0.8462	0.8845	0.9236	**0.8848**

The MELM model is thoroughly compared to other existing approaches in terms of various metrics in Table 4 and Figs. 3–4. With an accuracy of 62% and an F-score of 62.11%, the KNN model displayed limited detection, according to the results of dataset 1 analysis. Similar results were shown by the NB model, which had an F-score of 69.4% and a slightly better accuracy of 65%. The C5.0 model then attained an even higher accuracy of 73% and an F-score of 80.87%. The BLR model then produced an F-score of 72.43% and a moderate accuracy of 75%. The k-kNN model also earned an F-score of 77.92% and improved accuracy of 77%. The alignment result from the NN model was reasonable, with an F-score of 76.54% and an accuracy of 78%. The GBM model simultaneously displayed acceptable performance with an accuracy of 79% and an F-score of 79.17%. Additionally, although the DL model showed competitive accuracy of 83% and an F-score of 75.93%, the RF model produced near-optimal accuracy of 81% and an F-score of 82.19%. Notably, the MELM model that was presented demonstrated the best detection performance, achieving an F-score of 84.62% and an accuracy of 84%.

Table 4 Result Analysis of Existing with Proposed MELM Method

Methods	**F-Measure (%)**			**Accuracy (%)**		
	Dataset_1	**Dataset_2**	**Dataset_3**	**Dataset_1**	**Dataset_2**	**Dataset_3**
kNN	62.11	67.44	74.74	62.00	67.00	74.00
GBM	79.17	80.65	80.53	79.00	81.00	81.00
C5.0	80.87	87.48	91.90	73.00	72.00	72.00
NN	76.54	78.78	73.76	78.00	82.00	85.00
BLR	72.43	71.10	70.69	75.00	80.00	80.00
RF	82.19	82.24	88.07	81.00	88.00	92.00
NB	69.40	69.10	71.07	65.00	68.00	68.00
k-kNN	77.92	82.26	85.73	77.00	78.00	79.00
DL	75.93	80.10	82.16	83.00	88.00	92.00
MELM	84.62	88.45	92.36	84.00	88.25	92.25

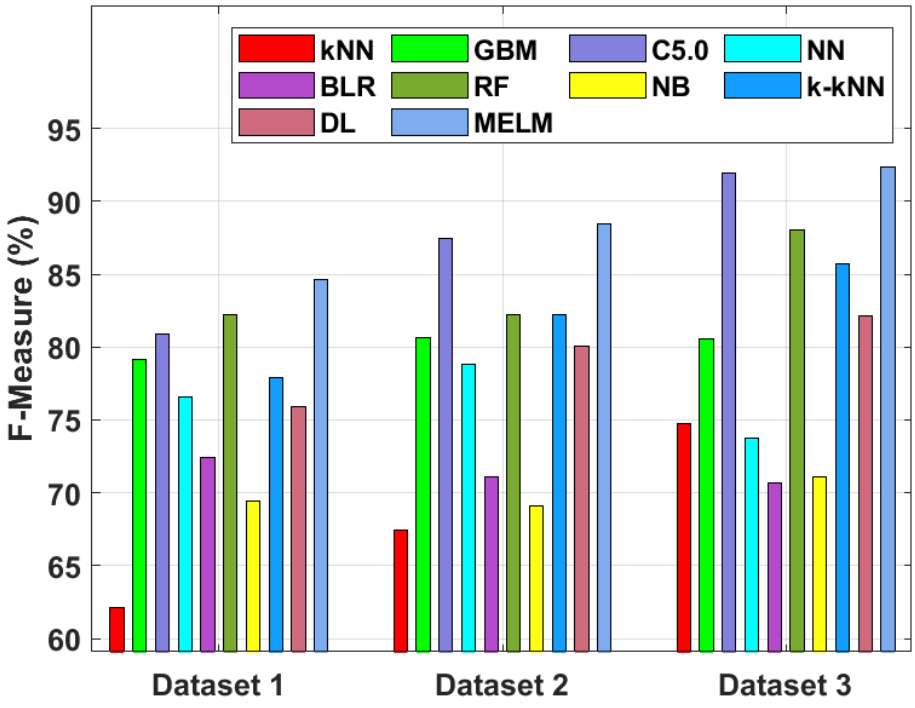

Fig. 3. MELM model F-measure analysis using current techniques

With a minimum accuracy of 67% and an F-score of 67.44%, the KNN approach showed limited detection in the dataset 2 analysis findings. Similar results were seen using the NB technique, which had an F-score of 69.10% and a slightly higher accuracy of 68%. In contrast, the C5.0 model got a noteworthy F-score of 87.48% and a high accuracy of 72%. The k-kNN model also produced an F-score of 82.26% and a reasonable accuracy of 78%. Likewise, the BLR method displayed a marginally higher accuracy of 80% and an F-score of 71.1%. The accuracy and F-score of the GBM methodology's results were also reasonable at 81% and 80.65%, respectively. The NN model also produced acceptable results, with an accuracy of 82% and an F-score of 78.78%. Additionally, the RF model showed competitive accuracy of 88% and an F-score of 82.24%, while the DL model showed better accuracy of 88% and an F-score of 80.1%. The proposed MELM model, however, outperformed previous approaches with a stunning F-score of 88.45% and a greater detection accuracy of 88.25%.

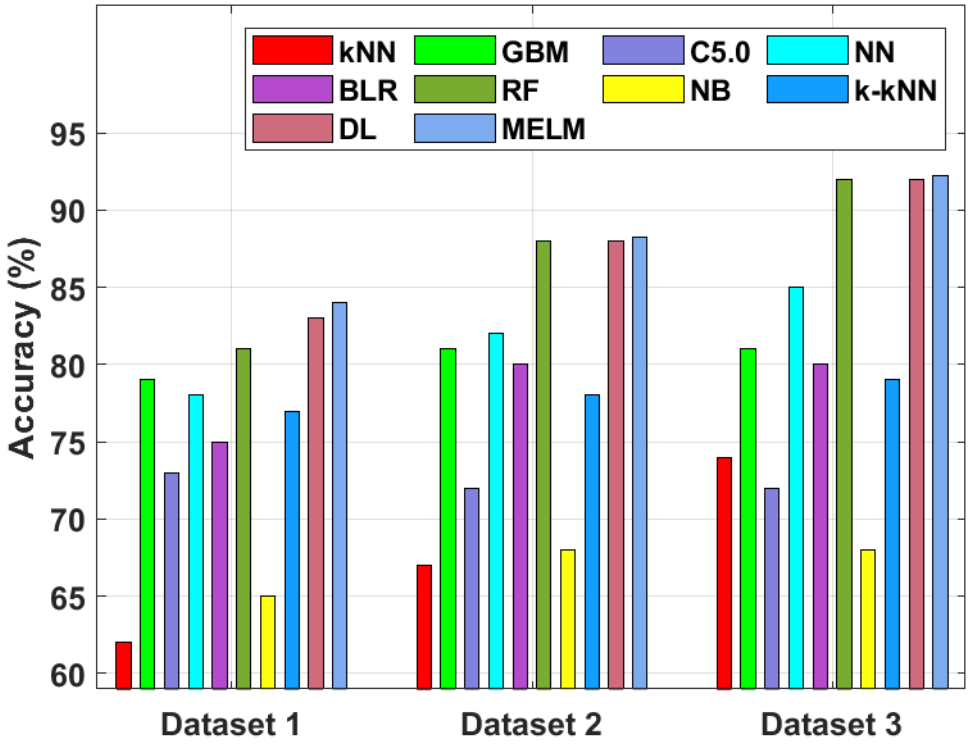

Fig. 4. MELM model accuracy analysis using existing techniques

Last but not least, the results of dataset 3 analysis showed that the NB technique had a restricted capacity for detection, with a minimum accuracy of 68% and an F-score of 71.07%. In contrast, the C5.0 method demonstrated increased accuracy, reaching 72%, along with an amazing F-score of 91.9%. Similar to this, the F-score and accuracy of the kNN approach were also 74.74%. The k-kNN model then produced fair results, with an F-score of 85.73% and an accuracy of 79%. The F-score and accuracy rates for the BLR approach were 80% and 70.69%, respectively. Accordingly, the GBM model produced acceptable results, with an F-score of 80.53% and an accuracy of 81%. The NN model achieved 85% accuracy and an F-score, which were satisfactory results. Additionally, the DL technique performed almost perfectly, with an F-score of 82.16% and an accuracy of 92%. Similar results were seen with the RF approach, which had an F-score of 88.07% and an accuracy of 92%. In the end, the proposed MELM approach surpassed all, with a stunning F-score of 92.36% and a superior detection accuracy of 92.25%.

5. Conclusion

This study introduces a novel Twitter spam detection model employing MELM. The model begins by taking in raw Twitter data, which is then subjected to preprocessing for enhanced data quality. Word2Vec is subsequently utilized to transform words in the dataset into multi-dimensional vectors. The MELM model extends the actual ELM structure by adding a few hidden layers, initializing weights between input and first hidden layers, along with bias for the first hidden layer. The output weights of the network are determined through the least squares technique. Evaluating the MELM model's enhanced spam detection efficiency, extensive experiments were conducted on three spam datasets. Results revealed that the MELM model achieved significant spam detection effectiveness, with accuracy, precision, recall, and F-Score values of 0.8817, 0.9057, 0.8650, and 0.8848 respectively.

References

[1] Benevenuto F, Magno G, Rodrigues T, et al. Detecting spammers on twitter. Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference (CEAS); Vol. 6; 2010. p. 12.

[2] A.-Z. Ala'M, J. Alqatawna, H. Faris, Spam profile detection in social networks based on public features, in: 2017 8th International Conference on information and Communication Systems (ICICS), IEEE, 2017, pp. 130–135.

[3] A.K. Ameen, B. Kaya, Detecting spammers in twitter network, Int. J. Appl. Math. Electron. Comput. 5 (4) (2017) 71–75.

[4] F. Benevenuto, G. Magno, T. Rodrigues, V. Almeida, Detecting spammers on twitter, in: Collaboration, electronic messaging, anti-abuse and spam conference (CEAS), 6, 2010, p. 12.

[5] K. Lee, J. Caverlee, S. Webb, Uncovering social spammers: social honeypots+ machine learning, in: Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval, ACM, 2010, pp. 435–442.

[6] C. Chen, J. Zhang, X. Chen, Y. Xiang, W. Zhou, 6 million spam tweets: A large ground truth for timely twitter spam detection, in: 2015 IEEE international conference on communications (ICC), IEEE, 2015, pp. 7065–7070.

[7] G. Stringhini, C. Kruegel, G. Vigna, Detecting spammers on social networks, in: Proceedings of the 26th annual computer security applications conference, ACM, 2010, pp. 1–9.

[8] A.H. Wang, Don't follow me: Spam detection in twitter, in: 2010 international conference on security and cryptography (SECRYPT), IEEE, 2010, pp. 1–10

[9] M. Fazil, M. Abulaish, A hybrid approach for detecting automated spammers in twitter, IEEE Trans. Inf. Forensics Secur. 13 (11) (2018) 2707–2719.

[10] C. Yang, R. Harkreader, G. Gu, Empirical evaluation and new design for fighting evolving twitter spammers, IEEE Trans. Inf. Forensics Secur. 8 (8) (2013) 1280–1293.

[11] T. Wu, S. Liu, J. Zhang, Y. Xiang, Twitter spam detection based on deep learning, in: Proceedings of the australasian computer science week multiconference, ACM, 2017, p. 3.

[12] X. Ban, C. Chen, S. Liu, Y. Wang, J. Zhang, Deep-learnt features for twitter spam detection, in: 2018 International Symposium on Security and Privacy in Social Networks and Big Data (SocialSec), IEEE, 2018, pp. 208–212.

[13] S. Lai, L. Xu, K. Liu, J. Zhao, Recurrent convolutional neural networks for text classification, Twenty-ninth AAAI conference on artificial intelligence, 2015.

[14] L. Mou, H. Peng, G. Li, Y. Xu, L. Zhang, and Z. Jin, "Discriminative neural sentence modeling by treebased convolution," arXiv preprint arXiv:1504.01106, 2015 Apr 5.

[15] K.S. Tai, R. Socher, and C.D. Manning, "Improved semantic representations from tree structured long short-term memory networks," arXiv preprint arXiv:1503. 00075, 2015 Feb 28.

[16] Xiao, D., Li, B. and Mao, Y., 2017. A multiple hidden layers extreme learning machine method and its application. *Mathematical Problems in Engineering*, *2017*.

[17] Gulati, M. ., Yadav, R. K. ., & Tewari, G. . (2023). Physiological Conditions Monitoring System Based on IoT. International Journal on Recent and Innovation Trends in Computing and Communication, 11(4s), 199–202. https://doi.org/10.17762/ijritcc.v11i4s.6514

[18] Wang Wei, Natural Language Processing Techniques for Sentiment Analysis in Social Media , Machine Learning Applications Conference Proceedings, Vol 1 2021.

[19] Wang Wei, Natural Language Processing Techniques for Sentiment Analysis in Social Media , Machine Learning Applications Conference Proceedings, Vol 1 2021.