An Efficient Framework for Fake Profile Identification Using Metaheuristic and Deep Learning Techniques

W. Rose Varuna[1], K. Shalini[2], Maria Elna Akkalya Roy[3]

[1]Assistant Professor, Department of Information Technology, Bharathiar University, Coimbatore, Tamilnadu, India.
[2]Student (II M.Sc), Department of Information Technology, Bharathiar University, Coimbatore, Tamilnadu, India.
[3]Student (II M.Sc), Department of Information Technology, Bharathiar University, Coimbatore, Tamilnadu, India.

E.mail: [1]hvaruna@gmail.com, [2]shalukathirit@gmail.com, [3]akkalyaroym@gmail.com

Abstract:
The rise of fraudulent accounts is one of the most severe concerns in the digital age. Fake accounts have been judged particularly destructive to both OSN service providers and their clients, and if not caught early, they might become considerably more troublesome in the future. A user becomes a target for an attacker when he or she creates an OSN account. Fake accounts may trace a user's movements and encourage them to do things they shouldn't. This article gives a comprehensive description of the framework for detecting false profiles in social networks. Account vulnerability assessment is monitored using an open-source big data system for false profile detection. In terms of public and private profile traits, there are ethical issues in data collecting.The LSTM configuration used to implement the proposed framework is also detailed. The suggested method has a training accuracy of 98 percent and a validation accuracy of 97.9%.

Keywords: Fake profile, classification, feature selection, meta-heuristic technique, deep learning, and LSTM.

1. INTRODUCTION

A fake user can evade detection techniques by learning the defense patterns employed by social network providers. It is also difficult to link the identity of a user across social networking platforms [1, 2]. The design of the proposed approach involves feature-based detection, considering low-level features along with many valuable social features [3, 4]. Tumblr, Twitter, Google+, Facebook, Instagram, Myspace, LinkedIn, and Foursquare are just a few of the numerous social networking services available. According to data, 823 million individuals utilised Facebook on their portable devices on a daily basis, which is hiked from 654 million in the preceding quarter [5, 6]. Facebook cannot provide real-time notifications about phoney profiles, and distinguishing between actual as well as fake profiles is tough for non-technically knowledgeable people. Furthermore, in order to get reliable profile identification outcomes, various big data challenges, such as streaming data, storage of data, and fast user replies, must be addressed when operating on enormous amounts of social big data [7]. According to the report filed by Facebook to the Securities Exchange and Commission, 8.7% of 955 million of its users are fakes. By now, 8.7% could have increased with the increase in the number of Facebook users [9, 10].

Most of the accounts classified as bogus (4.8%) are duplicates created by the user for maintaining personal and professional accounts separately with different names, which is a violation of the social network's policies [11, 12]. The attackers (1.5%) collect user details from different social networks and create a fake profile with collected attributes. These attackers mainly target non-technically savvy users, teens, females and children. The problem of determining a tight mapping between genuine users and online identities needs to be solved [13, 14].

Contribution

The main contribution of the research work is

- To develop an optimization approach for selecting optimal features from the dataset and this optimization approach can handle the premature convergence.
- To develop neural network for minimizing the error during the process of training.

The remainder of the article is organized as follows: the related works are discussed in Section 2, the proposed methodology is detailed in Section 3, the results are illustrated in Section 4, and the article is concluded in Section 5.

Related Works

The authors of the research [15] presented COMPA as a technique for detecting hacked accounts on Twitter and Facebook. The authors employed statistical modelling to create a behavioural profile of users based on the features of their transmitted communications, and they computed the anomaly score using numerous similarity metrics such as n-gram analysis. The weights of the features in the dataset were determined using Sequential Minimal Optimization (SMO) by the authors. The authors of study [16] used clustering algorithms on the wall postings to detect spam campaigns on Facebook. Based on the content (pictures, movies, etc.) posted on their walls, they further examined each malicious account for the presence of compromised accounts.

A lot of studies have focused on detecting duplicated accounts on social media platforms. The Markov Clustering algorithm (MCL) was used by the authors in [17] to divide the Facebook network into smaller communities based on their similarities. All the profiles that are similar to the real profiles were gathered to determine the significance of the association in order to determine whether it is a clone or not. Another study [18] proposes a system with three components: an information distiller, a profile hunter, and a profile validator, to identify social network profile copying.A research published in [19] built a network of people on Tianya forum and Taobao online auction website based on their themes of interest. The author pruned the graph to create a sockpuppet network based on the users' writing styles (SPN). Finally, SPN has been used to identify bogus communities using various community detection approaches.

In [20] describes another strategy for detecting bogus accounts on a Hong Kong-based discussion forum. The technique is based on the total number of subjects one account has submitted and the number of replies received from other users. To find a sockpuppet pair, a detection score is calculated. The higher the score, the more likely two accounts are to be a fraudulent account pair. The authors in [21] described a fake account detection system for the Wikipedia network utilising natural language processing techniques based on linguistic data such as the user's punctuation count, quotation-count, and usage of capital or lowercase.

The existing fake profile detection approaches faces the premature convergence during the process of optimization may not consider the significant features.Occurrence of error or complex training process can influence the performance of fake profile classification accuracy. By considering the drawbacks in the existing approaches, the proposed framework is formulated that is discussed in subsequent section.

Proposed Methodology

The process of profile data acquisition and the classification of acquired profile data attained for fake profile detection that is discussed in this section.

Pre-processing

The information in the post is handled in every stage with the assistance of NLTK libraries and the process of examination is accomplished in this research. The information in the post is transformed to accessible and process able format that uses several pre-processing techniques namely punctuation, lemmatization, acronym handling and stop word removal. The flow of raw data taken from a social network to the final chosen attribute subset via data profiling

Feature Selection-Dispersive File Swarm Optimization

The swarming behaviour of flies hovering around food sources inspired from the Dispersive Flies Optimisation (DFO) technique. Various factors influence flies' swarming behaviour, and the existence of a danger may disrupt their convergence on the marker (or the optimum value). As a result of considering the creation of swarms over the marker, the suggested algorithm also considers the breaking or weakening of swarms. Occurrence of convergence issue is rectified by linear scaling technique. In other words, in Dispersive Flies Optimisation, the flies' swarming behaviour is made up of two tightly linked mechanisms: the generation of swarms and the breaking or weakening of those swarms. The update equations' method and mathematical formulation are described below. The vectors position is determined by,

$$a_x^{-t} = a_{x1}^{-t} + a_{x2}^{-t} + \cdots + a_{xD}^{-t} \qquad i = 1,2,3 \ldots..NF$$

where the time is indicated by t, problem space dimension is indicated by D, and the count of flies is indicated by NF.

The population generation is initialized by time t=0 with xth vector and yth component is indicated as,

$$a_{xd}^{0} = a_{min.d} + r(a_{max.d} - a_{min.d})$$

where random selection from the solution space is indicated as r, interval of unit is given as U(0,1), initialization of bounds in dimension d is indicated as x_{min} and x_{max} for lower and upper bound respectively.

The components of the position vectors are changed separately on each iteration, taking into consideration the component's value, the corresponding value of the best neighbouring fly (consider ring topology) with the best fitness, and the value of the best fly in the whole swarm: The effective fitness with best fly in the entire solution space is given as,

$$a_{xd}^{t} = a_{nb.d}^{t-1} + U(0,1) \times (a_{sb.d}^{t-1} - a_{xd}^{t-1})$$

where best fly in the neighbour is indicated as $a_{nb.d}^{t-1}$, and best fly in the swarm is indicated as $a_{sb.d}^{t-1}$ across the uniform distribution range of 0,1. The value of uniform distribution is estimated by the linear scaling technique that preserves the premature convergence.

A dynamic rule for updating flies' positions (supported by a social neighbouring network that informs this update) and transmission of the findings of the best found fly to other flies are the two main components of the algorithm. As previously indicated, the swarm is disturbed for a variety of causes; one of the good effects of such disruptions is the relocation of the disturbed flies, which may lead to the discovery of a more suitable location. An element of stochasticity is introduced to the updating process to account for this possibility. Individual components of the position vectors of flies are reset based on this if the random number, r, produced from a uniform distribution on the unit interval U

(0, 1) is smaller than the disturbance threshold, dt. This ensures that the otherwise perpetual stagnation over a predicted local minimum is disrupted proportionately.

Classification-Long Short Term Memory

LSTM is a diversified form of standard network in that it takes a sequence vi=(vi1,vi2,…..,viT) as input and the process of iteration is accomplished from it=1 to T that generates the following,

$$q_{ti} = \Phi(y_{qit} + W_{hi}h_{ti-1} + W_{at^{vt}})$$

where q is a vector encoding the concealed unit q = (q_1, q_2, ..., q_T). Bias vectors are the b terms (eg. y_q denotes bias of the hidden layer). The nonlinear function Φ varies depending on the situation, however in the case of a generic RNN, it is generally the adoption of element-wise sigmoid (σ), as shown following equation. When an LSTM is utilised, however, this is calculated with Φ.

The acquired features are passed to the first two layers whereas the values are elected randomly that minimizes the error. This process returns the output after processing the data in the preceding layers and probability is estimated by the softmax activation function. The RNN elect the data that is remembered or forgotten. At a sequence of time ti, the learning network are updated as,

$$i_{ti} = \sigma(W_{ai}a_{ti} + W_{hi}h_{ti-1} + y_i)$$
$$f_{ti} = \sigma(W_{af}a_{ti} + W_{hf}h_{ti-1} + y_f)$$
$$g_{ti} = \tanh(W_{ac}a_{ti} + W_{hc}h_{ti-1} + y_c)$$
$$c_{ti} = i_{ti} \odot g_{ti} + f_{ti} \odot c_{ti-1}$$
$$o_{ti} = \sigma(W_{ao}a_{ti} + W_{hi}h_{to-1} + y_o$$
$$h_{ti} = o_{ti} \odot \tanh(c_{ti})$$

where the activation function is signified as σ, the multiplication with element wise is signified as ⊙, the input vector value at the time ti is signified as a_{ti}, the hidden states in the network is signified as h_{ti}, weight matrix for the diverse layers are signified as W_{ai}, W_{af}, W_{ac}, W_{ao} for a_{ti}, weight matrix for the diverse layers are signified as W_{hi}, W_{hf}, W_{hc}, W_{ho} for h_{ti}, the bias in the layer is signified as y_i, y_f, y_c, y_o, the input layer is signified as i, forgotten layer is signified as f, memory layer is signified as c and the output layer is signified as o. LSTM memory cells are effective in identifying and exploiting long range context.

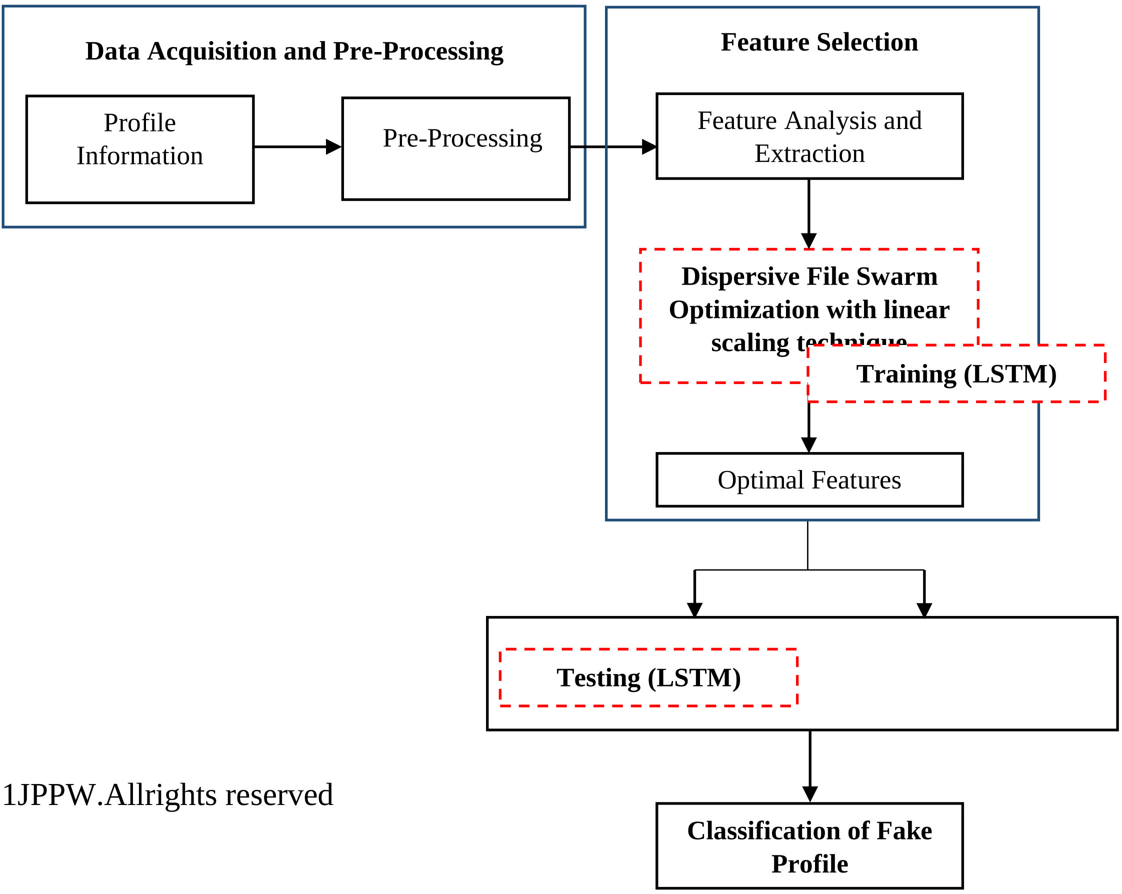
Data Acquisition and Pre-Processing
Profile Information
Pre-Processing
Feature Selection
Feature Analysis and Extraction
Dispersive File Swarm Optimization with linear scaling technique
Training (LSTM)
Optimal Features
Testing (LSTM)
Classification of Fake Profile

Figure 1. Overall block diagram of proposed methodology

2. RESULT AND DISCUSSION

The information deposited in SOCIAL NETWORKs is the user-made content and metadata of the users' profiles. The essential public profile information such as user name, screen name, surname, gender, zip code, country, education, hometown and workplace provide personal details of the user's identity. In addition, unique communication information, for instance email identities, phone numbers and personal profile information's are also taken as features. The connectivity using this information is characterized as the links in the social network graph that reflects various interests of the user. All together, the amount of personal information, which is provided directly by the social network user, constitutes diverse types of publicly available data. The process of obtaining this reliable data from streaming social network servers is significant. The proposed approach is compared with existing techniques namely SMO [15] and MCL [16] whereby the performance metrics such as accuracy, precision and recall is utilised for investigating their performance.

Accuracy: The classification accuracy of the fake profile is calculated by dividing the number of appropriate fake fake profile identifications by the total number of fake profile. Comparison of accuracy is given in Table 1 and Figure 2. The estimation of the accuracy is given as,

$$Accuracy = \frac{True\ Positive\ (TP) + True\ Negative(TN)}{True\ Positive(TP) + True\ Negative(TN) + False\ Positive(FP) + False\ Negative(FN)}$$

Table 1. Comparison of fake profile Classification Accuracy

S.No	Iteration	SMO	MCL	LSTM
1	50	85	83	95
2	100	85.5	83.5	96
3	150	86	85	96.5
4	200	87	86	98
5	250	88	87	98

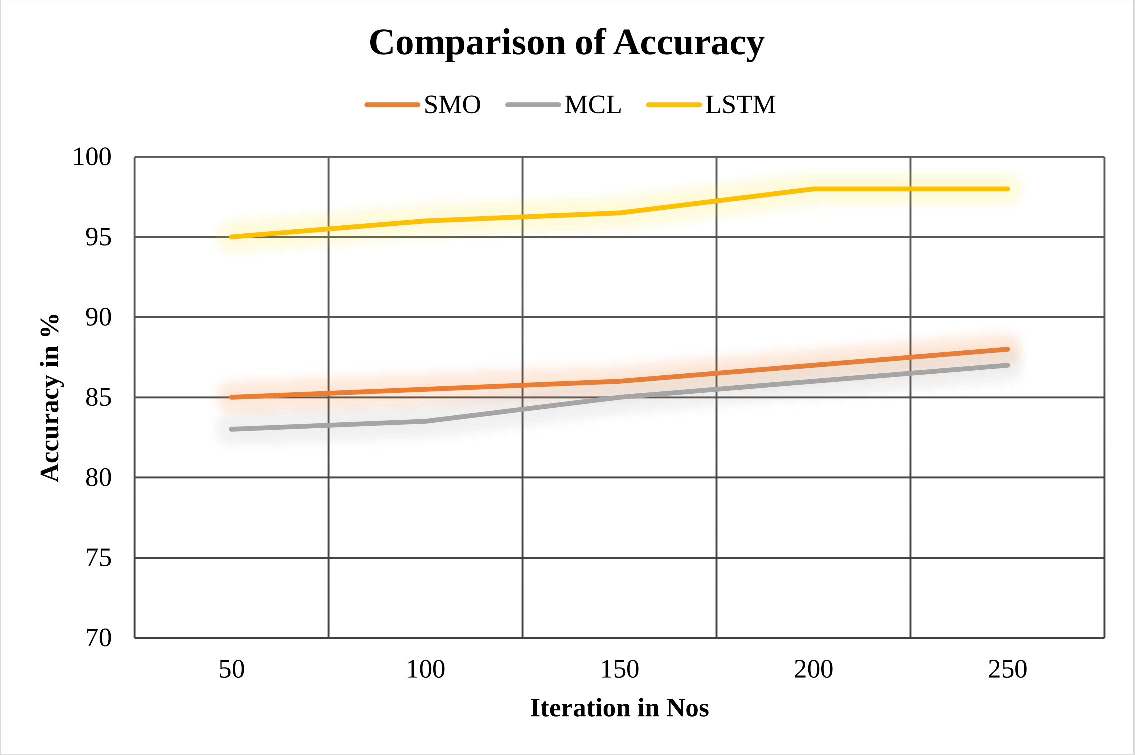

Figure 2. Comparison of fake profile Classification Accuracy

Precision: The quantitative rate with positive results, also known as precision, reflects the reliability of the prediction and the relevance of the feature found. Comparison of precision is given in Table 2 and Figure 3. It is equated as

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Table 2. Comparison of fake profile Classification Precision

S.No	Iteration	SMO	MCL	LSTM
1	50	81	86	96.4
2	100	82	87	97
3	150	86	88	97.4
4	200	86.4	88	97.8
5	250	84	89	98

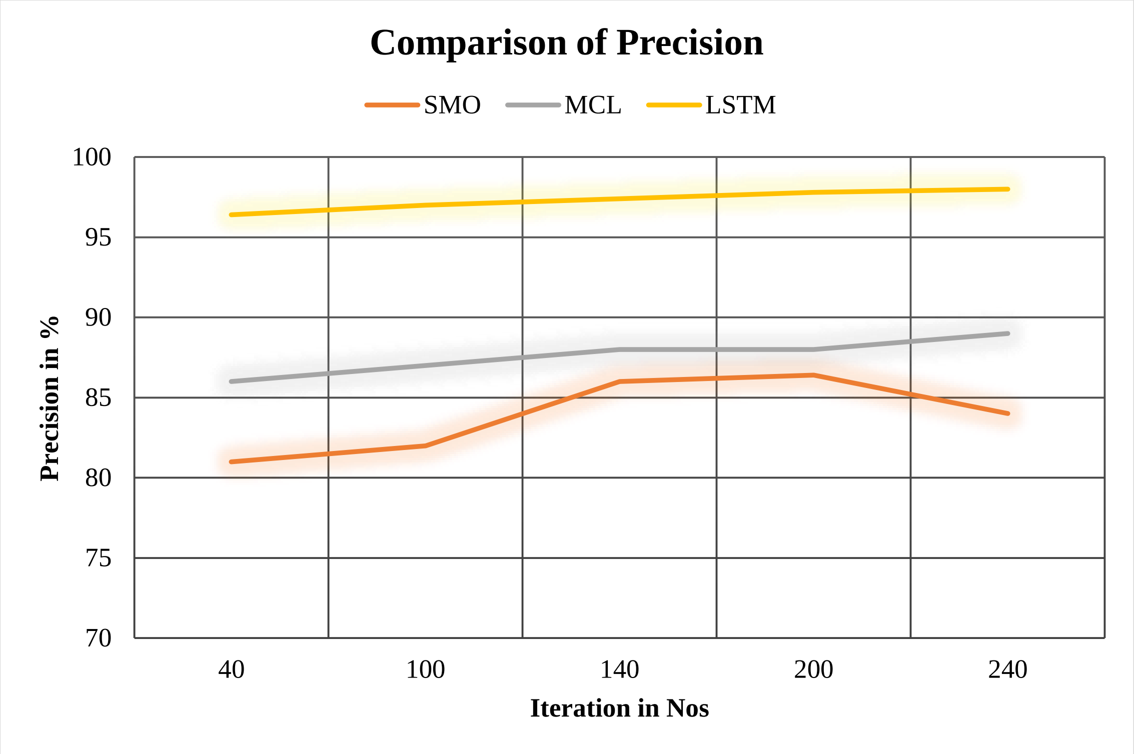

Figure 3. Comparison of fake profile Classification Precision

Recall: The associated fake profile among the substantially retrieved occurrences make up the rate of recall. Comparison of recall is given in Table 3 and Figure 4. It is calculated as

$$Recall = \frac{TP}{TP + FN}$$

Table 3. Comparison of fake profile Classification Recall

S.No	Iteration	SMO	MCL	LSTM
1	50	81	82	88
2	100	82	82	90
3	150	84	83	91
4	200	86	84	94
5	250	87	85	98

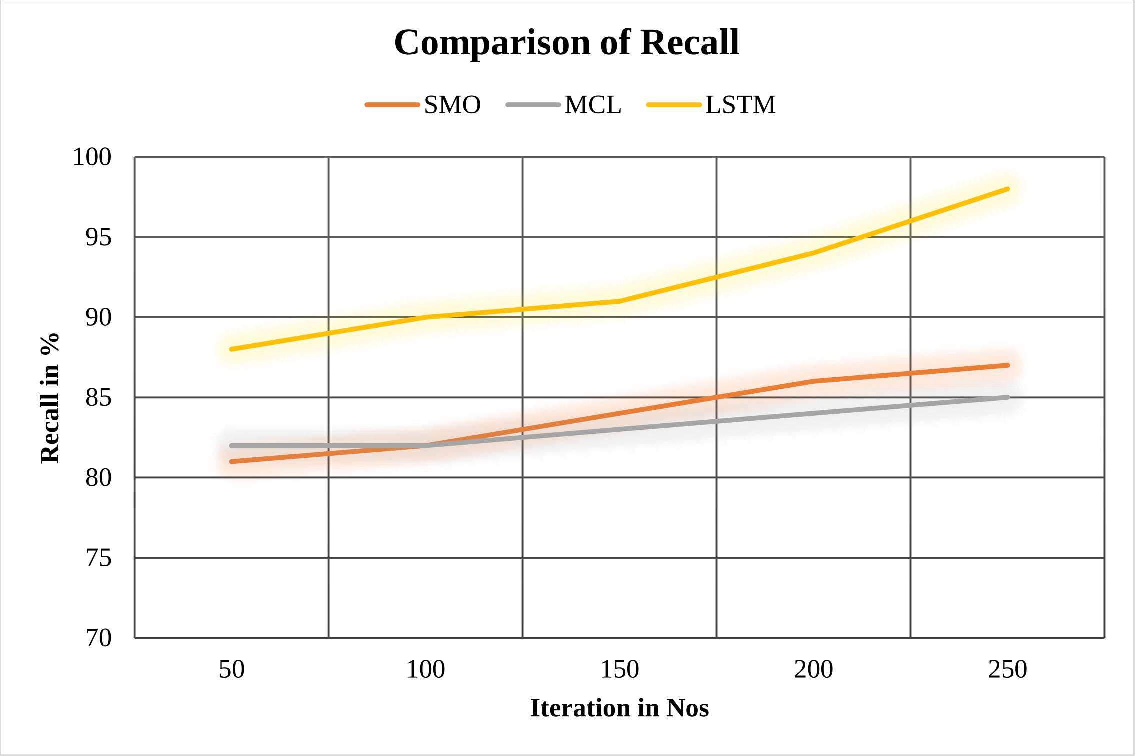

Figure 4. Comparison of fake profile Classification Recall

From the observation of the above tables and figures, it is identified that the proposed approach is highly effective where the performance of LSTM outperforms the existing approaches namely SMO as well as MCL.

3. CONCLUSION

One of the most serious issues in the digital era is the development of fake accounts. As they employ the OSN medium to perform everyday major crimes, cyber intelligence is attempting to relieve these profiles. Fake accounts have been deemed particularly harmful for both OSN service providers and their customers, and they might become much more problematic in the future if not discovered early. When a user opens an OSN account, he or she becomes a target for an attacker. Fake accounts can track a user's movements and persuade them to engage in illegal actions.This article provides a complete overview of the fake profile detection framework for identifying fake users in social networks. The open-source big data framework for fake profile detection is used to monitor account vulnerability assessment. The ethical considerations in data collection with respect to public and private profile attributes is explained. The LSTM setup for implementing the proposed framework is also explained in detail. The proposed approach attained 98% training accuracy and 97.9% validation accuracy.

4. REFERENCE

1. Roy, P. K., &Chahar, S. (2020). Fake profile detection on social networking websites: a comprehensive review. *IEEE Transactions on Artificial Intelligence*, *1*(3), 271-285.
2. Monti, F., Frasca, F., Eynard, D., Mannion, D., & Bronstein, M. M. (2019). Fake news detection on social media using geometric deep learning. *arXiv preprint arXiv:1902.06673*.
3. Liu, Y., & Wu, Y. F. B. (2020). Fned: a deep network for fake news early detection on social media. *ACM*

Transactions on Information Systems (TOIS), *38*(3), 1-33.

4. Mujeeb, S., & Gupta, S. (2022). Fake Account Detection in Social Media Using Big Data Analytics. In *Proceedings of Second International Conference on Advances in Computer Engineering and Communication Systems* (pp. 587-596). Springer, Singapore.
5. Khaled, S., El-Tazi, N., &Mokhtar, H. M. (2018, December). Detecting fake accounts on social media. In *2018 IEEE international conference on big data (big data)* (pp. 3672-3681). IEEE.
6. Lu, Y. J., & Li, C. T. (2020). GCAN: Graph-aware co-attention networks for explainable fake news detection on social media. *arXiv preprint arXiv:2004.11648*.
7. Awan, M. J., Khan, M. A., Ansari, Z. K., Yasin, A., &Shehzad, H. M. F. (2021). Fake profile recognition using big data analytics in social media platforms. *Int. J. Comput. Appl. Technol*.
8. Sansonetti, G., Gasparetti, F., D'aniello, G., &Micarelli, A. (2020). Unreliable users detection in social media: Deep learning techniques for automatic detection. *IEEE Access*, *8*, 213154-213167.
9. Elyusufi, Y., &Elyusufi, Z. (2019, October). Social networks fake profiles detection using machine learning algorithms. In *The Proceedings of the Third International Conference on Smart City Applications* (pp. 30-40). Springer, Cham.
10. Singh, N., Sharma, T., Thakral, A., & Choudhury, T. (2018, June). Detection of fake profile in online social networks using machine learning. In *2018 International Conference on Advances in Computing and Communication Engineering (ICACCE)* (pp. 231-234). IEEE.
11. Shu, K., Wang, S., & Liu, H. (2018, April). Understanding user profiles on social media for fake news detection. In *2018 IEEE conference on multimedia information processing and retrieval (MIPR)* (pp. 430-435). IEEE.
12. Kumar, E. Boopathi, and V. Thiagarasu. "Comparison and Evaluation of Edge Detection using Fuzzy Membership Functions." *International Journal on Future Revolution in Computer Science & Communication Engineering (IJFRCSCE), ISSN*: 2454-4248.
13. E. B. Kumar and V. Thiagarasu, "Color channel extraction in RGB images for segmentation," *2017 2nd International Conference on Communication and Electronics Systems (ICCES)*, 2017, pp. 234-239, doi: 10.1109/CESYS.2017.8321272.
14. Sahoo, S. R., & Gupta, B. B. (2021). Real-time detection of fake account in twitter using machine-learning approach. In *Advances in computational intelligence and communication technology* (pp. 149-159). Springer, Singapore.
15. Yang, S., Shu, K., Wang, S., Gu, R., Wu, F., & Liu, H. (2019, July). Unsupervised fake news detection on social media: A generative approach. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 33, No. 01, pp. 5644-5651).
16. Ahvanooey, M. T., Zhu, M. X., Mazurczyk, W., Choo, K. K. R., Conti, M., & Zhang, J. (2022). Misinformation Detection on Social Media: Challenges and the Road Ahead. *IT Professional*, *24*(1), 34-40.
17. Egele, M., Stringhini, G., Kruegel, C., &Vigna, G. (2013, February). Compa: Detecting compromised accounts on social networks. In *NDSS*.

18. Gao, H., Hu, J., Wilson, C., Li, Z., Chen, Y., & Zhao, B. Y. (2010, November). Detecting and characterizing social spam campaigns. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (pp. 35-47).
19. Kharaji, M. Y., &Rizi, F. S. (2014). An iac approach for detecting profile cloning in online social networks. *arXiv preprint arXiv:1403.2006*.
20. Kontaxis, G., Polakis, I., Ioannidis, S., &Markatos, E. P. (2011, March). Detecting social network profile cloning. In *2011 IEEE international conference on pervasive computing and communications workshops (PERCOM Workshops)* (pp. 295-300). IEEE.
21. Bu, Z., Xia, Z., & Wang, J. (2013). A sock puppet detection algorithm on virtual spaces. *Knowledge-Based Systems*, *37*, 366-377.
22. Zheng, X., Lai, Y. M., Chow, K. P., Hui, L. C., &Yiu, S. M. (2011, October). Sockpuppet detection in online discussion forums. In *2011 Seventh International Conference on Intelligent Information Hiding and Multimedia Signal Processing* (pp. 374-377). IEEE.
23. Solorio, T., Hasan, R., &Mizan, M. (2013, June). A case study of sockpuppet detection in wikipedia. In *Proceedings of the Workshop on Language Analysis in Social Media* (pp. 59-68).
24. Yookesh, T. L., et al. "Efficiency of iterative filtering method for solving Volterra fuzzy integral equations with a delay and material investigation." *Materials today: Proceedings* 47 (2021): 6101-6104.
25. Kumar, E. Boopathi, and V. Thiagarasu. "Segmentation using Fuzzy Membership Functions: An Approach." *IJCSE, ISSN* (2017): 2347-2693.