AN INTELLIGENT FAKE PROFILE CLASSIFICATION SYSTEM USING DEEP LEARNING TECHNIQUES

W. Rose Varuna
Assistant Professor, Department of Information Technology,
Bharathiar University, Coimbatore, Tamilnadu, India.
hvaruna@gmail.com
L. Prasanth
Student (II M.Sc), Department of Information Technology,
Bharathiar University, Coimbatore, Tamilnadu, India.
prasanthlingan@gmail.com
A. Pabitha
Student (II M.Sc), Department of Information Technology,
Bharathiar University, Coimbatore, Tamilnadu, India.
Pabitha1929@gmail.com

Abstract: Fake accounts can be created by humans, computers, or even cyborgs. A cyborg account is half-human and half-bots. A person physically initiates the account, but a bot handles all subsequent activities. There are differences between bots versus human profiles. When profiles are fake and not hacked by authorized users, bots are referred to as Sybil profiles. The research is recommended using a Twitter dataset, which consolidates real and Fake profile and implements that dataset using various methods like machine learning, natural language processing, and deep learning. Using these datasets, examination of distinct exploratory analyses to distinguish semantic properties broadly present in unreliable content. Furthermore correlate our Fake profile identification standards precision with expected accuracy to put our consequences in aspect. Utilization of Natural Language Processing, machine learning, and deep learning procedures to complete our models and distinguish which models will give higher efficiency. The CNN has a training accuracy of 99 percent accuracy.

Keywords: Fake profile, feature selection, classification, deep learning, fake user, ICA, and CNN.

1. Introduction

The social networking sites like Facebook, Twitter is the most significant ways of internet communication and collaboration [1, 2]. Fake information spreading by bots is another major problem of social media, around 30% of posted information is fake or bogus every day from malicious web applications or bots. So, it is important to improve the trustworthiness of social media by detecting Fake profile and bots timely. Analyzing user's profiles information and identifying its trustworthiness using various soft computing techniques is the way of eliminating Fake profile distribution [3-5]. To commit numerous cybercrimes such as profile hacking, identity hacking, session hijacking, malicious linking, mail bombs, and so on helps criminals build false identities.

Mostly bots or humans may create such kinds of fake identities. In general, the fake identities of bots target large numbers of individuals on social media [6]. Fake information or accounts can spread forged information rapidly without any verification policy, which is the big drawback of social media [7]. Identity deception on various social media platforms has become a growing problem with the tremendous increase in the number of accounts on these platforms. Attackers have used fake identities for several malicious purposes, which are created by bots and humans. This system removes accounts by bots from the corpus during preprocessing and performs classification of accounts by humans into two categories, i.e., Fake vs. Real using Recurrent Neural Network (RNN) algorithm based on different parameters [8, 9].

A bot is a computer program that performs a specific task over the Network; it is also called an internet robot, WWW robot, or just a simple bot. Bots usually complete basic and conceptually repetitive activities, though greater rate than just a human being or a single entity might be able to do [10-12]. The biggest use of bots is in the collection of data sources, in that a dynamic script extracts, evaluates, and files virtual server content at several times the higher frequency of a human. Bots specifically generat Fake profile with massive data uploading rate [13]. Fake profile is inaccurate knowledge generated through business activity to gather awareness and generate promotion revenue or spread negativity violations to have a political influence.

News stories suggest truthfulness but include purposeful mistakes of fact with the anticipation of exciting interests, attracting audiences, or cheating. There have already been lots of instances of unapprovable or unauthorized false data circulating rapidly completed informal online entities since late [14]. For example, there have been ongoing allegations of Russian electronic Network hacking in Virginia and reports indicating that Saudi Arabia funds the presidential campaign of Emmanuel Macron. Since about late, such unverified news has

DOI: 10.9756/INT-JECSE/V14I3.361

circulated rapidly, so it is challenging to channel certain news only with the production of huge datasets in these areas [15].

This work also demonstrates a feature of Fake profile detection that classifies the news article as trustworthy or fake. Investigation of fake profile identification using various models and classifiers and predict unconventional models and classifiers' efficiency. It will check which prototype will give more precision and incorporate the news into real or fake. It also produce computational sources and illustrations for the work of Fake profile apprehension [16].

1.1. Contribution

The main contribution of the research work is

- To develop an optimization approach for selecting optimal features from the dataset using Independent Component Analysis (ICA).
- To develop Convolutional Neural Network for minimizing the error during the process of training that is evaluated using accuracy.

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.

2. Related Works

According to Estee et al. [17] trained the classifier by applying used features for bot detection in order to identify fake accounts created by the human on Twitter. The training is based on supervised learning. They have tested for three different classifiers, i.e., Support Vector Machine (SVM) with linear kernel, Random Forest (RF) and Adaboost. For SVM classification, the SVM linear library in R software is used. Here the boundary based on feature vectors is created for classification. For the RF model, the RF library in R software is used. RF model creates variations of trees, and mode of class outcome is used to predict identity deception. For boosting model, the Adaboost function in R is used. Adaboost is used along with decision trees where each feature is assigned a different weight to predict the outcome. These weights are iteratively adjusted, and output is evaluated for the effectiveness of identity deception prediction at each iteration. This process is repeated until the best result is obtained. Among these three classifiers, RF reached the best result.

Information from user posts on two common social media platforms was collected in [18] template files: Twitter and Facebook. A user's depression level was identified based on his social media messages. Structured research or quasi-interview procedure (SDI) [17] is the traditional way of identifying a person's depression. Such approaches require a tremendous amount of knowledge from the individual. Mobile messaging sites such as Twitter and YouTube have become widespread platforms for sharing people's activities and opinions. Statistical analysis from tweets and posts shows the presence of the user's symptoms of major depression. Quantum computing is used in this study to process the discarded data obtained from SNS users. Natural Language Processing (NLP), categorized to diagnose depression more conveniently and accurately using the SVM and Naïve Bayes algorithm.

According to [19], the interest in identifying false profiles and researching their activities has increased. Questions such as who the impersonators are? What are their distinctive features? And are they robots? It is going to emerge. To react, the framework begins this research by gathering data on Instagram from three significant groups, including "Politician," "News Agency," and "Sports Star." Four top checked accounts are selected within each group. The device detects 4K based on the users who replied to their published posts.

The authors of the research [20] 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.

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 [21] 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.

Occurrence of redundant and replicated feature can influence the classification. 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.

3. 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.

3.1. Pre-processing

DOI: 10.9756/INT-JECSE/V14I3.361

The information in the post is handled at each level with the help of NLTK libraries, and the inspection procedure is completed in this study. Several pre-processing approaches are used to convert the information in the post into a legible and executable format, including punctuation, lemmatization, acronym management, and stop word removal. Data profiling is used to move raw data from a social network to a final attribute subset.

3.2. Feature Selection-Independent Component Analysis

Independent Component Analysis (ICA) is based on the theory of instantaneous linear aliasing blind signal separation. The output of the approximate independent source signals are statistically independent of each other. The Fast ICA algorithm is based on the criterion of non-Gaussian maximization, which uses a fixed-point recursive algorithm. To find the optimal value of the cost function and makes the separated signals that have the best mutually independent values. A fixed-point algorithm is identified based on kurtosis, and a fixed-point algorithm based on negative entropy is developed.

Negative entropy is used as a measure of non-Gaussian signal, and its convergence effect is better than kurtosis. But it is difficult to calculate the theoretical value of negative entropy in practice. An optimal algorithm solves the approximate maximum value of negative entropy. The Newton iterative algorithm finds the optimal separation matrix OPS_{max} by batch processing to continuously update the weight WI, so that the negative entropy reaches the maximum.

Choose the number of iterations p and the number of the component needed to estimate m, the steps of Fast-ICA algorithm are as follows:

Algorithm 1. Independent Component Analysis (ICA)

Centralize the acquired information A and the mean value is zero where the value Z is replaced by A

Estimation of e, the count of the necessary components are selected and the count of the iteration is assigned as t

Initial phase is randomly elected W_t and ordered on the basis of

$$W_t = E\left\{Z_g\left(\frac{W}{Z}\right)\right\} - E\left\{g'\left(\frac{W}{Z}\right)\right\}W$$

Newton Iterative Approach is incorporated and the values are gathered P-I

$$W_t = W_t - L(W_t^T W_j)W_j$$

Regularization of data is accomplished and if the convergence is met then step 3 is repeated

The value of p is assigned as p=p+1 and if p<m, then step 2 is returned

3.3. Classification-Convolutional Neural Network

The process of classification process is attained using self-structured feed forward simple CNN and it is detailed in this section.

Convolutional Layer

It examines the features of a profile information with its neighboring features as they are strongly correlated as compared to those features that are distant. It would be worthwhile to analyze the influence of neighbourhood data. In CNN, this influence is learned by convolutional layer. The dimension of the resultant feature maps, and after convolution, depends on several parameters (that is discussed in subsequent sections). For N number of filters in each layer of CNN, N activation maps are acquired. These activation maps are handled depth wise (or stacked together), in order to, get the final output for single convolutional layer (comprises of N filters). Dimension of resultant feature maps/activation maps depends on following parameters:

Padding (P): Due to convolution, original profile information dimension is lost (i.e. it shrinks). Further, edge features, that contain vital information, are used less number of times. In view of this, original profile information is padded with features of zero values around its borders. Broadly speaking, main concern of padding is to ensure that the dimension of activations maps should be equal to input profile information

Stride (S): It gives number of shifts for a kernel to take while convolving an profile information. For example, with stride 2 filter is shifted 2 features during convolution. Consider n □ n is the size of original profile information, f □ f is the size of the kernel, P□1 is the padding, and S □ 2 is the stride then the dimension of a resultant feature map, after applying one filter, is computed as:

$$\left\lfloor\frac{n+2P-f}{s}+1\right\rfloor \times \left\lfloor\frac{n+2P-f}{s}+1\right\rfloor$$

For N filters in a convolutional layer, total dimension of the feature maps is N times the equation 1. Generally, odd and equal dimension for f is used.

Pooling Layer

It progressively down samples the spatial size of feature maps, hence, reduces total number of parameters and overall computation in the network. For each feature map pooling is applied individually and equal sub-area wise. There are various types of pooling, however, max-pooling is commonly used, which extracts that value in each sub-area whose magnitude is maximum. These functions also have the specifications

DOI: 10.9756/INT-JECSE/V14I3.361

for stride that is used for controlling the dimension of the outputs. It aid in to reduce the translation variance. Besides max-pooling other forms are also used, such as, average pooling, and weighted average pooling, etc.

Fully Connected Layer

Present at the end in a CNN, receives the flattened (vectorised) output of last pooling or convolutional layers. Each neuron in this layer is fully connected to all activations of the previous layers. Due to this fully connection classification of features or activations are carried out. Note, this layer is not needed for semantic segmentation because spatial context is lost in this layer.

Learning Properties

Learning of the convolutional neural network depends on estimating a misfortune work additionally called target work, mistake work, cost work) that demonstrates the blunder of educated network parameters. The learning target is to figure the parameters to limit the misfortune work. Softmax work (Equation 4) is the likelihood of class e_i given information X, where y_i speaks to the score for i^{th} class among absolute e classes. The softmax misfortune E is determined as negative log probability of the softmax work (Equation 5), where N indicates the length of the class vector.

$$p(c_i X) = \frac{e^{zi}}{\sum_{j=i}^{c} e^{zi}}$$

$$E = \left(\frac{-1}{N}\right) \sum \quad log\left(P\left(\frac{c_n}{X}\right)\right)$$

The technique utilized for advancing the misfortune minimization is called Stochastic Gradient Descent with Momentum. For the t^{th} cycle, the update procedure is indicated by,

$$\varphi_t = v\varphi_{t-1} - \nabla G(w_{t-1})$$
$$w_t = w_{t-1} + \varphi_t$$

where φ_t indicates the present weight update, φ_(t-1) is the past weight update, w speaks to the loads, G is the normal misfortune over the dataset, rG(w) is negative slope, is the learning rate and 2 [0; 1] utilized for accelerating the combination of inclination and anticipating motions. The overall proposed methodology is given in Figure 1.

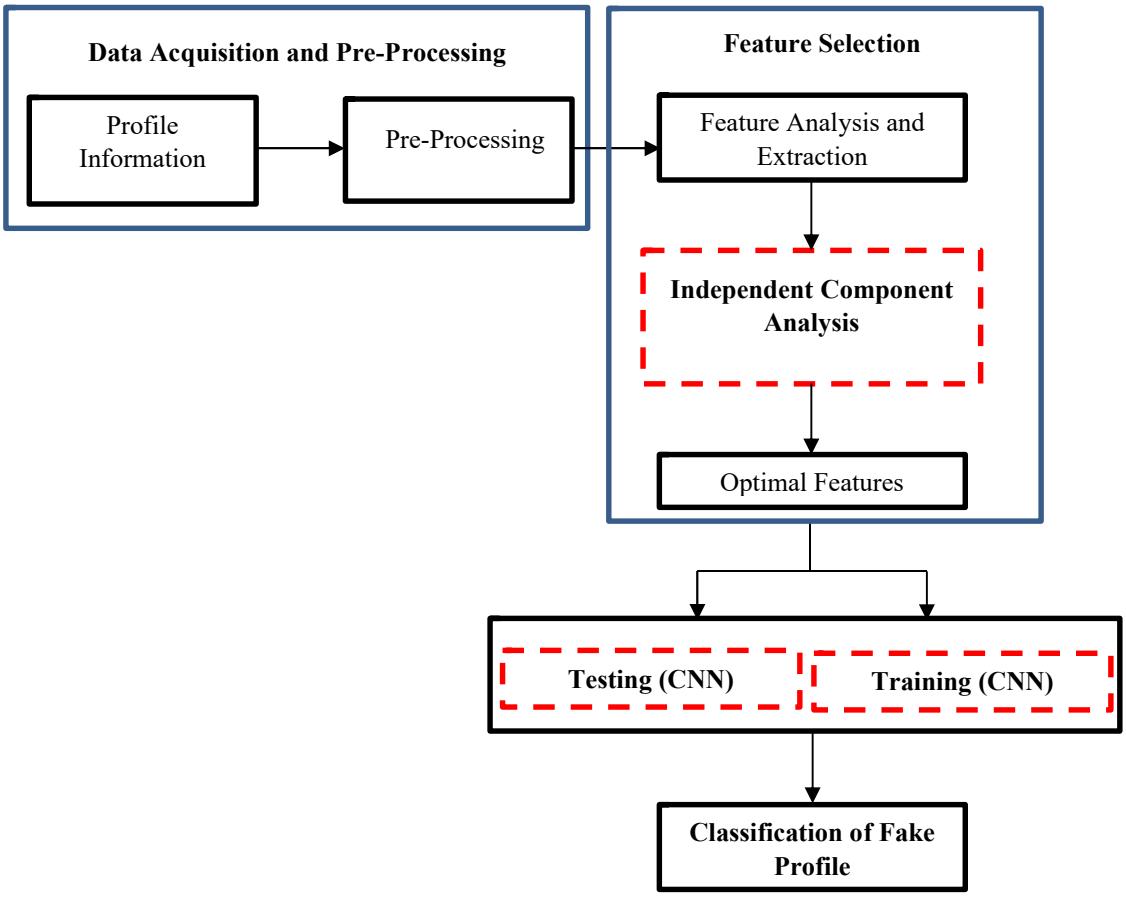

Figure 1. Overall block diagram of proposed methodology

DOI: 10.9756/INT-JECSE/V14I3.361

4. Result and Discussion

The user-created content and metadata of the users' profiles are the information placed in social networks. User name, screen name, surname, gender, zip code, nation, education, birthplace, and workplace are all important public profile facts that reveal the user's identify. Unique communication data, such as email IDs, phone numbers, and personal profile information, is also considered a feature. The relationships in the social network graph that represent the user's diverse interests are described as connectedness utilising this information. The quantity of personal information that is directly shared by the social network user adds up to a variety of publicly available data. Obtaining this trustworthy data from streaming social network servers is a time-consuming task. The proposed approach is compared with existing techniques namely SMO [20] and MCL [21] 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	CNN
1	50	85	83	96
2	100	85.5	83.5	96.5
3	150	86	85	97
4	200	87	86	98
5	250	88	87	99

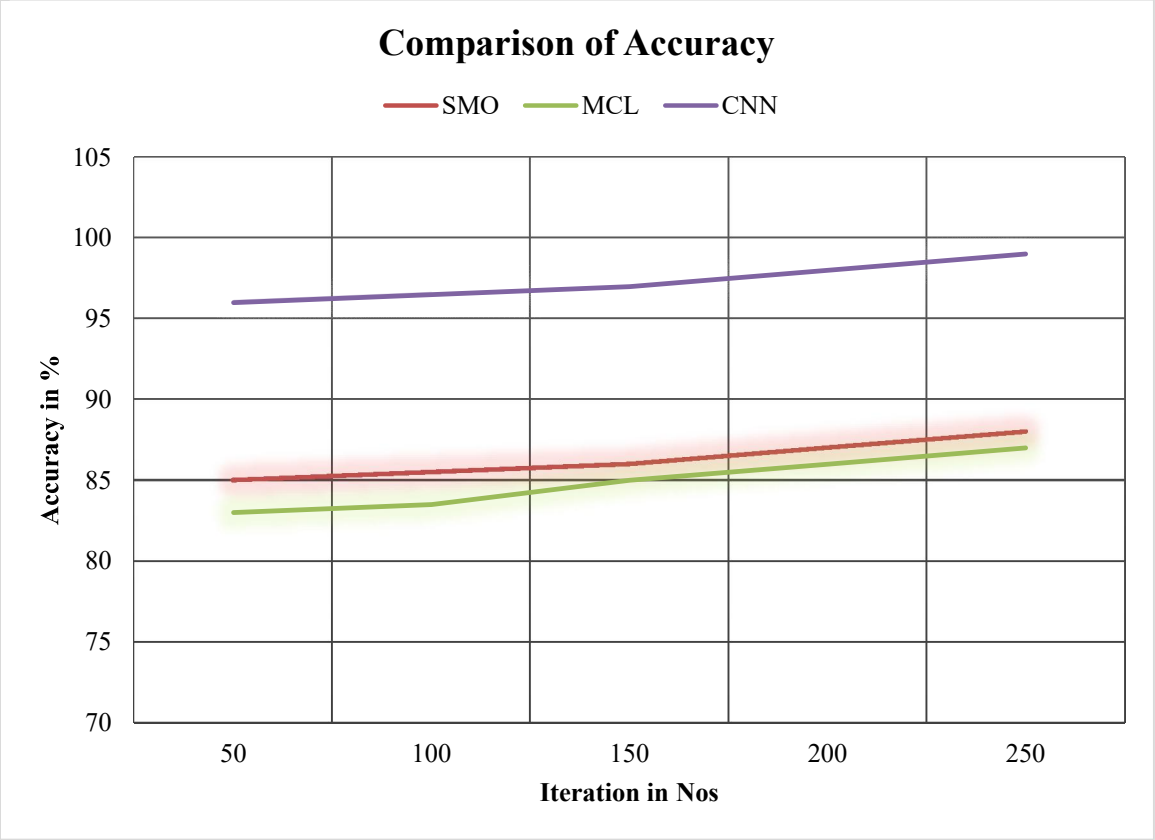

Figure 2. Comparison of fake profile Classification Accuracy

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

DOI: 10.9756/INT-JECSE/V14I3.361

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

Table 2. Comparison of fake profile Classification Precision

S.No	Iteration	SMO	MCL	CNN
1	50	81	86	96
2	100	82	87	96.5
3	150	86	88	97
4	200	86.4	88	98
5	250	84	89	98

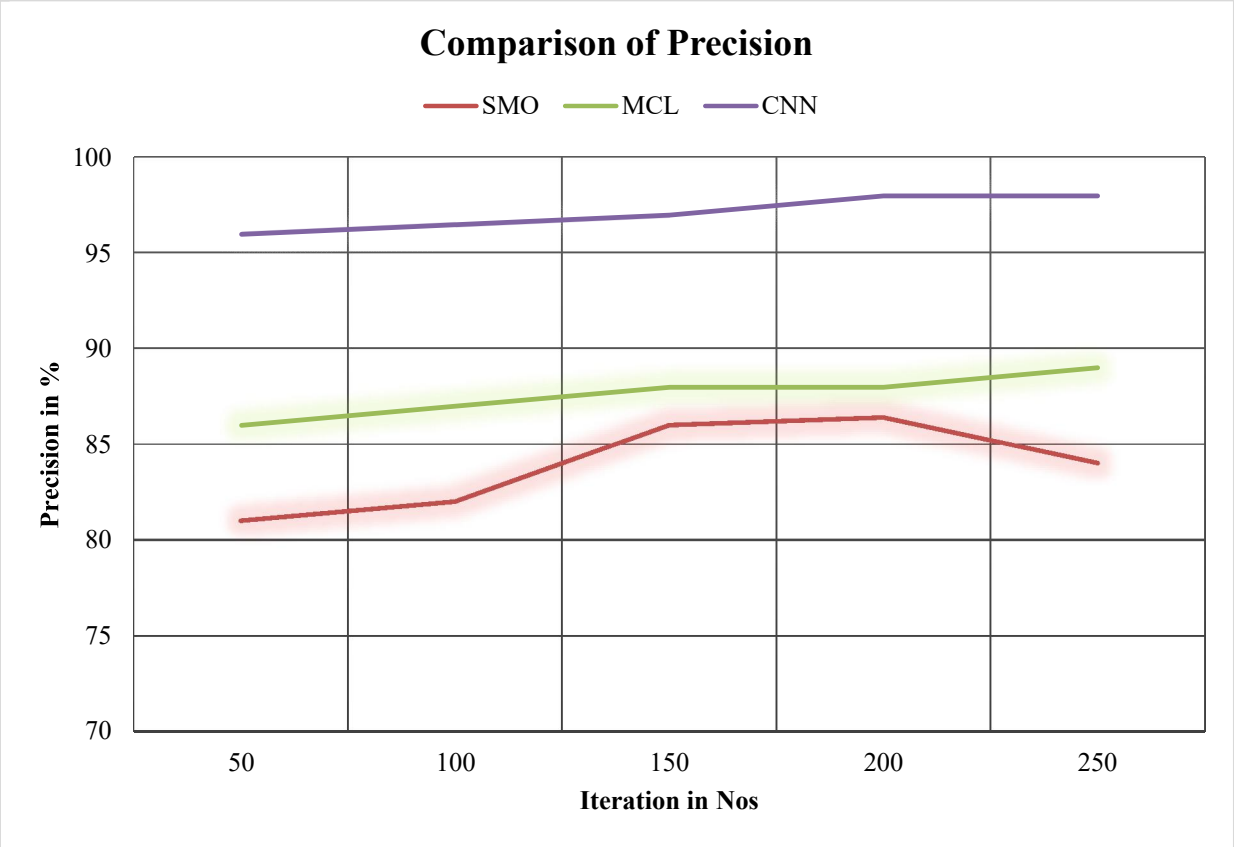

Figure 3. Comparison of fake profile Classification Precision

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	CNN
1	50	81	82	89
2	100	82	82	89.5
3	150	84	83	90
4	200	86	84	95
5	250	87	85	97

DOI: 10.9756/INT-JECSE/V14I3.361

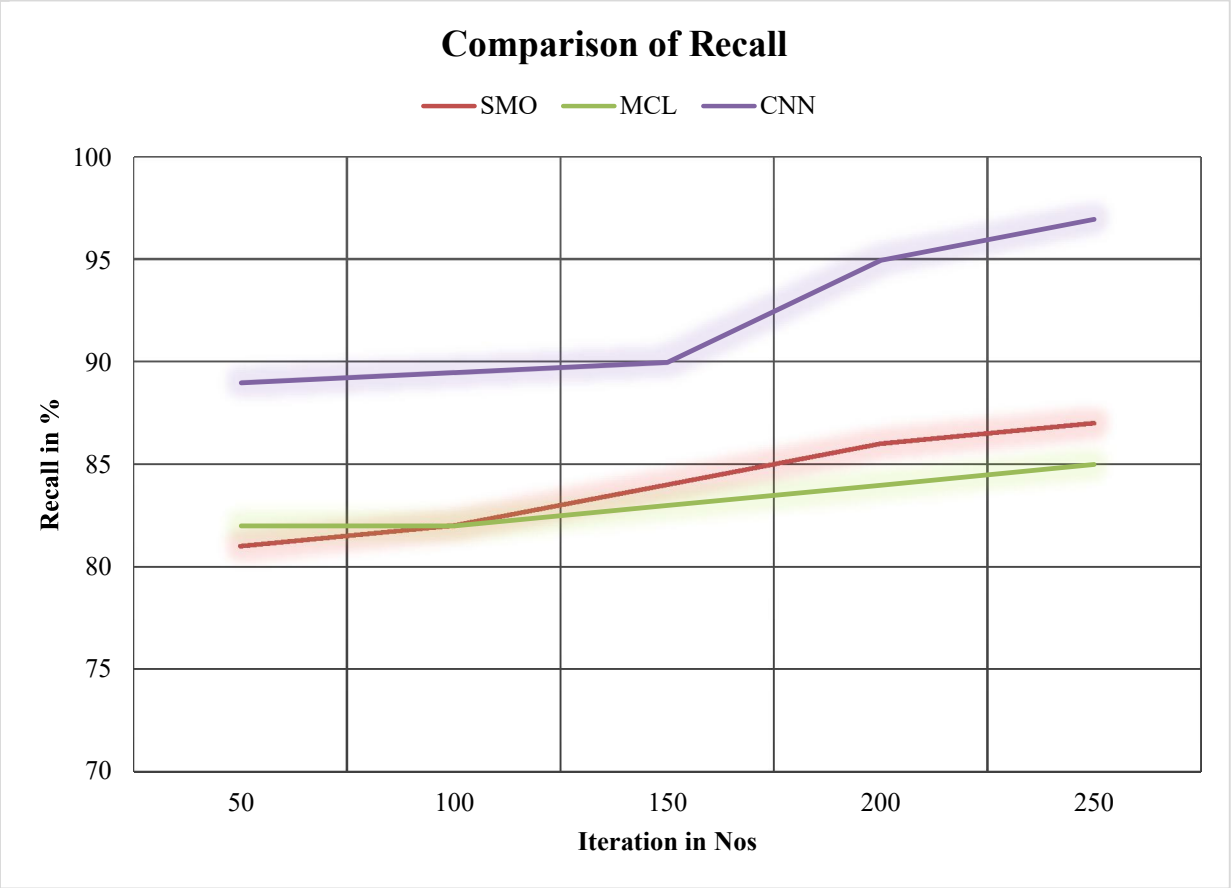

Figure 4. Comparison of fake profile Classification Recall

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

5. Conclusion

Humans, machines, and even cyborgs may establish fake accounts. A cyborg account is made up of half-humans and half-machines. The account is created by a person, but all subsequent operations are handled by a bot. There are several distinctions between bot and human profiles. Bots are referred to as Sybil profiles when profiles are fraudulent and not hacked by authorised users. The study recommends employing a Twitter dataset, which combines actual and fake profiles and uses machine learning, natural language processing, and deep learning to create the dataset. Examining several exploratory approaches using these datasets to determine semantic features often present in untrustworthy material. To put our outcomes in perspective, compare the precision of our Fake profile identification standards to the predicted accuracy. To complete our models and determine which models will provide greater efficiency, we used Natural Language Processing, machine learning, and deep learning techniques. The CNN has a 99 percent training accuracy rate.

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*.

DOI: 10.9756/INT-JECSE/V14I3.361

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. 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.
9. Kumar, E. Boopathi, and V. Thiagarasu. "Segmentation using Fuzzy Membership Functions: An Approach." *IJCSE, ISSN* (2017): 2347-2693.
10. 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.
11. 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.
12. 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.
13. 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.
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. Bhattacharya, A., Bathla, R., Rana, A., & Arora, G. (2021, September). Application of Machine Learning Techniques in Detecting Fake Profiles on Social Media. In *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)* (pp. 1-8). IEEE.
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. Van Der Walt, E., & Eloff, J. (2018). Using machine learning to detect fake identities: bots vs humans. *IEEE access*, *6*, 6540-6549.
20. Shahane, P., & Gore, D. (2018). A Survey on Classification Techniques to Determine Fake vs. Real Identities on Social Media Platforms.
21. Kondeti, P., Yerramreddy, L. P., Pradhan, A., & Swain, G. (2021). Fake account detection using machine learning. In *Evolutionary computing and mobile sustainable networks* (pp. 791-802). Springer, Singapore.
22. Egele, M., Stringhini, G., Kruegel, C., & Vigna, G. (2013, February). Compa: Detecting compromised accounts on social networks. In *NDSS*.
23. Kharaji, M. Y., & Rizi, F. S. (2014). An iac approach for detecting profile cloning in online social networks. *arXiv preprint arXiv:1403.2006*.
24. 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.
25. 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.

DOI: 10.9756/INT-JECSE/V14I3.361