IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

FAKE PROFILE DETECTION USING DEEP LEARNING

Ms.B. Gunasundari, Mr Barath L, Mr Harigaran K, Mr Hariharan V

Associate Professor, Student, Student, Student

Department Of Computer Science And Engineering

Prathyusha Engineering College, Chennai, India

Abstract: Online social networks (OSN) have greatly improved communication, information exchange, and enjoyment in modern society. However, because of their accessibility and anonymity, OSNs have provided a favorable environment for a variety of harmful practices like spamming, trolling, fake news, and astroturfing. One of the primary threats to OSNs is socialbots, which are computer programs that perform various illicit activities. To address the threat of socialbots, researchers have been developing various detection methods. One of the latest and most advanced methods is SBRidAPI, which stands for SocialBot RID (Rapid Identification) using deep API learning. The goal of SBRidAPI is to detect socialbots by analyzing a user's behavior on OSNs.SBRidAPI models a broad range of profile, temporal, activity, and content information for user behavior representation using deep learning techniques. Profile information includes user's name, age, location, and other similar data, whereas temporal information is about the frequency and timing of user activities. Activity information includes the type of activities, such as likes, comments, and shares. Content information includes the text, images, and videos shared by the user. SBRidAPI represents profile, temporal, and activity information as sequences in order to analyze the sequential nature of this information that is supplied to a two-layers stacked BiLSTM. Deep CNN is fed content data in order to analyze the text content and learn the visual characteristics of images and videos. Once SBRidAPI has analyzed a user's behavior, it assigns a score that reflects the likelihood of the user being a socialbot. The user is labeled as a socialbot if their score is higher than a threshold that is then used to compare scores. SBRidAPI is the first method that jointly models a complete collection of profile, temporal, activity, and content information for user behavior representation, making it an effective tool for identifying socialbots on OSNs.

Index Terms **- Deep learning, LSTM, CNN, SBRidAPI**

I. INTRODUCTION

Social media platforms are widely used for communication and information sharing. However, some users exploit these platforms for malicious activities, such as spreading fake news, phishing attacks, and promoting hate speech. Many of these activities are performed through social media bots, which are automated accounts pretending to be human users. Detecting these social bots is challenging, as they can mimic human behavior to evade detection. Traditional approaches to bot detection rely on analyzing large amounts of social media posts and network structures, but they may not be effective in identifying individual bot accounts.This study suggests a deep learning-based strategy called SBRidAPI that employs a combination of feature engineering, behavior modeling, and deep neural networks to identify social bots as a solution to this problem. To reflect user behavior, this model incorporates a wide range of user profile, temporal, activity, and content data. SBRidAPI models profile, temporal, and activity information as sequences using a two-layer stacked BiLSTM and a deep CNN to model content information. Additionally, an attention mechanism is incorporated to highlight important features in the input data. SBRidAPI is capable of learning contextual representations of each behavioral aspect of users, which allows it to detect bots created through automation tools by analyzing user profile attributes and tweet times. The model is designed to classify individual social media accounts from single observations, such as a single tweet, without relying on before data of users behavior. The results show that SBRidAPI can accurately detect social bots from a single tweet and outperforms previous works that rely on user profiles and recent posting history.

II. RELATED WORKS

David Martín-Gutiérrez et al. have proposed a multilingual approach to identify suspicious Twitter accounts using deep learning techniques. The authors have created a system that, regardless of the language used in the account, is able to identify suspicious accounts based on a set of criteria. They have used deep learning techniques to generate user embeddings, and then applied new approaches to produce non-parametric data automatically Finding differences and resemblances between the distributions of both bots and non-bot embeddings using two samples of data. The proposed multilingual approach can be helpful in identifying bots across different languages, which is especially useful as Twitter is a global platform used by people from various countries and cultures.

Abdelouahid Derhab et al. have reviewed different machine learning techniques to distinguish between human and bot accounts on Twitter. They have provided a taxonomy that categorizes the most recent machine learning methods for detecting tweet-based bots. Social media platforms, including Twitter, have become popular targets for malicious activities such as spamming, phishing, and spreading fake news. As a result, it is now possible to identify between human and tweet-based bot accounts with accuracy by using large data analytics tools, particularly shallow and deep learning approaches. This can serve as a useful resource in machine learning-based tweet-based bot detection techniques.

To solve imbalances in Twitter bot identification, Bin Wu and Le Liu have suggested a data augmentation method utilizing conditional generative adversarial networks. The authors have avoided the creation of data-augmentation noise and eliminated imbalances between and among social bot class distributions by using a modified clustering approach, the Gaussian kernel density peak clustering algorithm. Additionally, they have added the Wasserstein distance with a gradient penalty to address the traditional CGAN model collapse and gradient disappearance and improve the convergence judgment condition. The suggested approach may be useful in addressing the issue of unbalanced datasets in Twitter bot detection and may result in more precise bot account detection. The use of generative adversarial networks can also provide a more efficient and effective way of data augmentation.

Octavio Loyola-González proposed the usage of a contrast pattern-based classifier for bot detection in Twitter. The contrast pattern-based classification and Random Forest produced the best classification results when they compared the output of 21 classifiers. However, compared to the patterns identified by the suggested method, Random Forest produced a significantly higher number of rules.

Peining Shi employs transition probability features between user clickstreams based on social situation analytics to detect malicious social bots on social network platforms in real-time. They designed an algorithm for detecting malicious social bots based on spatiotemporal features using a semi-supervised clustering method to reduce the time of artificial marking. Their tests demonstrated that it is possible to accurately identify dangerous social bots using the transition likelihood between user clickstreams based on social scenario analytics.

III. PROPOSED SYSTEM

This paper presents SBRidAPI, to profile users for detecting socialbots on OSNs. To the best of our knowledge, this is the first deep learning-based approach that jointly models a comprehensive set of profile, temporal, activity, and content information for user behavior representation. It models profile, temporal, and activity information as sequences, which are fed to a two-layers stacked BiLSTM, whereas content information is fed to a deep CNN.

3.1 SBRidAPI APP

To simulate and classify a user using and browsing an OSN, in this module develop a SBRidAPI can use web automation, which includes methods for creating and populating multiple OSN accounts, tweets, crawling the social graph and executing online social activities.

3.2 Data Set Acquisition

To train and test our model, we collect all public datasets of labeled human and bot accounts and create three new ones, all available in the bot repository MIB ang Kaggle.

3.3 Preprocessing

Prior to training the Bi LSTM and CNN on the dataset, preprocess the data by forming a string of tokens from each tweet. 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 normalization 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 normalization methods, such as lemmatization and stemming, can be used to identify the word root forms.

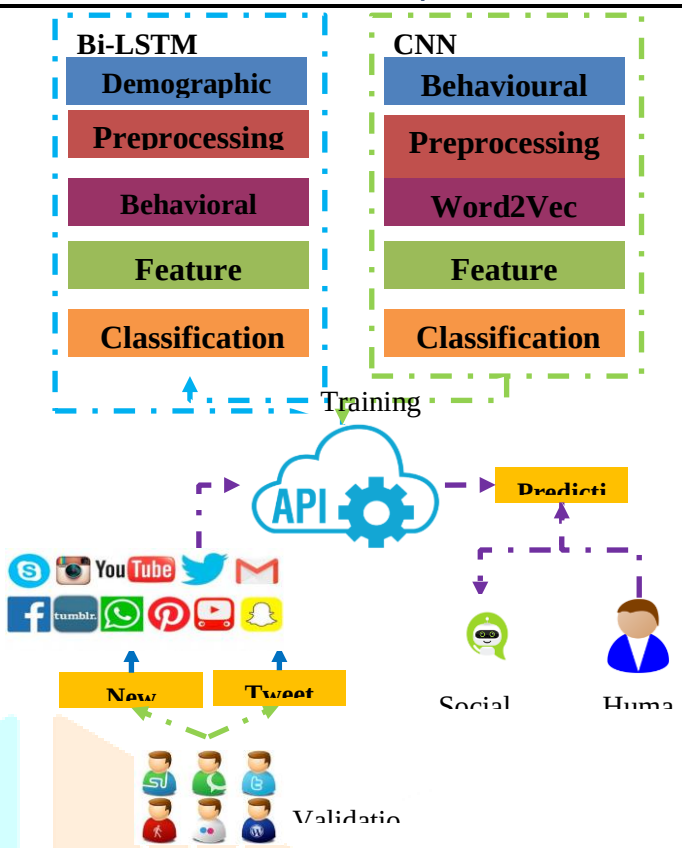

fig. 3.1 Architecture Diagram

.

3.4 Feature Extraction

For building a machine learning classifier we require a matrix of feature values for training and later on - classifying unseen test data. So, in a first step we extract a variety of features from the datasets. These features are discussed in the next subsections. 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.

3.5 Classification

This module uses two classification tasks: account-level bot detection and tweet-level bot detection. when OSN data is pre-processed and Feature Extracted, BiLSTM-CNN model is used in the classification of users into either legitimate users or bots.

3.6 Decision Model

In this module it combines all the features and embeddings together and create SBRid Model for final prediction. Obtain the classified normal user set and social bots set: the normal user set and social bots set can be finally obtained by detecting with live OSN user data.

IV. RESULTS AND DISCUSSION

The study used common metrics for evaluating text classification methods such as precision, recall, F1, and accuracy. The accuracy was chosen as the most suitable measure to assess the effectiveness of the method since the dataset was balanced. Based on the type of content each account created—human-made, CNN-based neural network produced content, BiLSTM-based deep learning produced content, and groups of people—the accounts in the dataset were divided into four groups. The study acknowledged that they could have improved the dataset by excluding links from tweets, but this would have reduced the size of the dataset significantly. Additionally, gathering more tweets per user would have provided more training data for the model, but it would have taken too long to implement.

V. CONCLUSION

This paper introduces a model called SBRidAPI for detecting socialbots on online social media platforms. Socialbots are automated accounts that can be used for malicious purposes like launching attacks and manipulation campaigns. SBRidAPI uses a combination of BiLSTM and CNN models to analyze users' behavior and detect malicious bots accurately. It doesn't require any prior knowledge about the user's profile or behavior. This makes it faster and easier to implement and deploy for bot detection. The proposed model can also be adapted for other problems like detecting phishing emails or webpages by using BiLSTM with word embeddings.

REFERENCES

[1] Homsi, Ahmad, Joyce Al Nemri, Nisma Naimat, Hamzeh Abdul Kareem, Mustafa Al-Fayoumi, and Mohammad Abu Snober. "Detecting Twitter Fake Accounts using Machine Learning and Data Reduction Techniques." In DATA, pp. 88-95. 2021.

[2] Kumar, Ayush, and Teng Joon Lim. "Early detection of Mirai-like IoT bots in large-scale networks through sub-sampled packet traffic analysis." In Advances in Information and Communication: Proceedings of the 2019 Future of Information and Communication Conference (FICC), Volume 2, pp. 847-867. Springer International Publishing, 2020.

[3] Devle, Aniket Chandrakant, Julia Ann Jose, Abhay Shrinivas Saraswathula, Shubham Mehta, Siddhant Srivastava, Sirisha Kona, and Sudheera Daggumalli. "BotNet Detection on Social Media." arXiv preprint arXiv:2110.05661 (2021).

[4] M. Imran, M. H. Durad, F. A. Khan, and A. Derhab, ``Toward an optimal solution against denial of service attacks in software de_ned networks," *Future Gener. Comput. Syst.*, vol. 92, pp. 444_453, Mar. 2019.

[5] M. S. Savell. (2018). Protect Your Company's Reputation From Threats by Social Bots.

[6] Aslam, Salman. "Twitter by the numbers: Stats, demographics & fun facts." Omnicoreagency. com (2018).

[7] A. Aldweesh, A. Derhab, and A. Z. Emam, ``Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues," Knowl.-Based Syst., vol. 189, Feb. 2020, Art. no. 105124.

[8] S. Mahdavifar and A. A. Ghorbani, ``Application of deep learning to cybersecurity: A survey," Neurocomputing, vol. 347, pp. 149_176, Jun. 2019.

[9] E. B. Karbab, M. Debbabi, A. Derhab, and D. Mouheb, ``MalDozer: Automatic framework for Android malware detection using deep learning," Digit. Invest., vol. 24, pp. S48_S59, Mar. 2018.

[10] F. A. Khan, A. Gumaei, A. Derhab, and A. Hussain, ``A novel twostage deep learning model for ef_cient network intrusion detection," IEEE Access, vol. 7, pp. 30373_30385, 2019.

[11] A. Derhab, A. Aldweesh, A. Z. Emam, and F. A. Khan, ``Intrusion detection system for Internet of Things based on temporal convolution neural network and ef_cient feature engineering," Wireless Commun. Mobile Comput., vol. 2020, pp. 1_16, Dec. 2020.

[12] Marr, Bernard. "How twitter uses big data and artificial intelligence (ai)." Bernard Marr (2020).

[13] A. T. Kabakus and R. Kara, ``A survey of spam detection methods on Twitter," Int. J. Adv. Comput. Sci. Appl., vol. 8, no. 3, pp. 29_38, 2017.

[14] M. Chakraborty, S. Pal, R. Pramanik, and C. R. Chowdary, ``Recent developments in social spam detection and combating techniques: A survey," Inf. Process. Manage., vol. 52, no. 6, pp. 1053_1073, Nov. 2016.

[15] E. Alothali, N. Zaki, E. A. Mohamed, and H. Alashwal, ``Detecting social bots on Twitter: A literature review," in Proc. Int. Conf. Innov. Inf. Technol. (IIT), Nov. 2018, pp. 175_180.

[16] M. Latah, ``Detection of malicious social bots: A survey and a re_ned taxonomy," Expert Syst. Appl., vol. 151, Aug. 2020, Art. no. 113383.

[17] K. Yang, O. Varol, C. A. Davis, E. Ferrara, A. Flammini, and F. Menczer, ``Arming the public with arti_cial intelligence to counter social bots," Hum. Behav. Emerg. Technol., vol. 1, no. 1, pp. 48_61, Jan. 2019.

[18] Z. Guo, J.-H. Cho, I.-R. Chen, S. Sengupta, M. Hong, and T. Mitra, ``Online social deception and its countermeasures: A survey," IEEE Access, vol. 9, pp. 1770_1806, 2021.

[19] Abkenar, Sepideh Bazzaz, Mostafa Haghi Kashani, Mohammad Akbari, and Ebrahim Mahdipour. "Twitter spam detection: a systematic review." arXiv preprint arXiv:2011.14754 (2020).

[20] W. Daffa, O. Bamasag, and A. AlMansour, ``A survey on spam URLs detection in Twitter," in Proc. 1st Int. Conf. Comput. Appl. Inf. Secur. (ICCAIS), Apr. 2018, pp. 1_6.