Kerbala Journal of Computing and Technology 2025

Detecting Bot-Controlled Accounts on Social Media Using Deep Learning

Asia Mahdi Naser Alzubaidi[a], Noor Sabah Sagheer[b], Zahraa k.Asendia[c]*

[a]*Computer Science Department, College of Computer Science and Information Technology, Karbala University, Karbala, Iraq, Email: asia.m@uokerbala.edu.iq*

[b]*Computer Science Department, College of Computer Science and Information Technology, Karbala University, Karbala, Iraq, Email: noor.sabah@uokerbala.edu.iq*

[c]*Computer Science Department, College of Computer Science and Information Technology, Karbala University, Karbala, Iraq, Email: zahraa.k@uokerbala.edu.iq*

Received: 11 / 02 /2025
Accepted: 21 / 05 /2025
Published:30 / 06 / 2025

Keywords:

Bot Detection, Twitter Dataset

Deep Learning, Feature Engineering, Evaluation Matrices

A B S T R A C T

The proliferation of bot-controlled accounts on social media platforms poses significant risks to user trust and platform integrity. Recent approaches to bot identification suffer from imbalanced data, as well as overfitting and scalability issues. To address these challenges, this paper proposes a deep learning-based framework to detect such accounts using behavioral and content-based features extracted from Twitter data. The methodology integrates feature engineering, data preprocessing, and deep learning models. Evaluated on the Cresci-2017 dataset, the best-performing model achieved a test accuracy of 98.51%, with a precision of 99.16%, recall of 98.23%, and an F1-score of 98.69%. The results show that deep learning can effectively differentiate between genuine and bot-controlled accounts, contributing to enhanced security and authenticity in online interactions.

1. Introduction

Online Social Networks (OSNs) have transformed the way people interact, facilitating seamless communication and information exchange. However, alongside these benefits comes a growing threat—bot-controlled accounts that manipulate online discourse. These automated entities are often deployed to spread misinformation, amplify propaganda, influence public opinion, and engage in fraudulent activities [1]–[3]. Their ability to operate at scale makes them a significant challenge for maintaining the integrity of digital platforms.

Malicious bots have become increasingly sophisticated, exhibiting coordinated behavior and mimicking human-like activity to evade detection. They manipulate social interactions, alter profile features, and disguise themselves as legitimate users [4]. This deceptive nature makes identifying and eliminating bots a complex task. Compounding this challenge is the imbalance between genuine users and bots, as the latter usually exist in smaller numbers, creating difficulties for classification models that rely on balanced data distributions [5], [6].

Traditional bot detection approaches typically use rule-based techniques that rely on predefined behavioral patterns and handcrafted features [7]. While these methods have shown some effectiveness, they struggle to adapt to the

∗**Corresponding Author**: Asia Mahdi Naser Alzubaidi

Email address: *asia.m@uokerbala.edu.iq*

evolving strategies of bot developers. Static rule sets become outdated as bots modify their tactics, rendering conventional detection mechanisms less effective over time [8].

To address these limitations, deep learning has emerged as a promising solution for bot detection in OSNs. Unlike rule-based systems, deep learning models can automatically extract meaningful patterns from large datasets without requiring manually crafted features [9]. These models analyze multiple aspects of user behavior, including activity patterns, content characteristics, network structures, and temporal dynamics, to differentiate between human and automated accounts [10]. By leveraging deep learning, bot detection systems can become more robust, adaptive, and capable of mitigating the spread of misinformation and harmful automated activity on social media platforms [11].

2. **Related Works**

Several studies have explored bot detection using different methodologies. Early methods primarily relied on rule-based techniques, but recent advancements have shifted toward machine learning, deep learning approaches, and generative adversarial networks (GANs) for improved accuracy and adaptability [12]. Each approach has its strengths and limitations, with performance varying based on dataset characteristics and model architecture.

Cresci et al. (2017) examined the evolution of social spambots and highlighted a major shift in bot behavior, where automated accounts increasingly mimic human-like activity to evade detection [13]. Their study analyzed various bot detection techniques, including behavior-based and feature-engineered methods. Their findings highlighted the limitations of rule-based approaches, which struggle against increasingly sophisticated bots. The study underscored the need for adaptive detection models capable of handling evolving bot strategies.

Kudugunta and Ferrara (2018) introduced deep neural networks (DNNs) for bot detection, demonstrating that deep learning models significantly outperform traditional classifiers. Their study evaluated performance based on accuracy, precision, recall, and F1-score, achieving an accuracy of 95.3%, a precision of 96.1%, and an F1-score of 94.8%, demonstrating the effectiveness of deep learning in capturing complex behavioral patterns [14]. However, the study noted that deep models require substantial computational resources and large amounts of labeled training data, making real-time implementation challenging.

Yang et al. (2020) proposed GANBOT, a framework using Generative Adversarial Networks (GANs) for bot detection. Their approach involved training a generator to create bot-like profiles and a discriminator to differentiate real bots from genuine users [15]. This adversarial training improved model robustness against previously unseen bot behaviors. However, the GAN-generated samples sometimes failed to fully represent real-world bot behaviors, potentially affecting model generalization.

More recent works, such as Ellaky et al. (2024), introduced a hybrid BiGRU-LSTM model with GloVe word embeddings to enhance text-based bot detection. Their approach improved bot classification based on language usage patterns, achieving a precision of 97.2% [16]. Another study by Lingam and Das (2025) introduced a Variational GAN (VGAN) with Hidden Markov Models (HMMs) for bot detection in Twitter networks [17]. Their method effectively modeled temporal bot behavior, improving long-term detection accuracy. However, their approach required significant computational power, making it challenging to deploy on large-scale networks in real time.

3. Methodology

This study utilizes the Cresci-2017 dataset to detect bot-controlled accounts on Twitter. The dataset, annotated by CrowdFlower contributors, contains information about genuine users and automated bots. It includes key features such as followers count, friends count, statuses count, favorites count, and tweet activity [18]. The proposed system follows a structured approach to classify social media accounts as humans or bots. The methodology consists of several key steps:

3.1 Data Cleaning

Before applying machine learning techniques, the dataset was preprocessed to ensure data consistency and quality.

3.1.1 Adding Labels

- A new column was introduced to label accounts as either "genuine" (human) or "bot" (automated) to facilitate supervised learning.

3.1.2 Merging Dataframes

- Since the Cresci-2017 dataset consists of separate CSV files for genuine and bot accounts, these files were merged into a single unified dataframe.

3.1.3 Handling Missing Values

- Columns with excessive missing values were removed to maintain data quality.
- Remaining missing values were imputed using mean or median values where necessary.

3.2 Feature Engineering

To enhance classification accuracy, meaningful features were extracted, focusing on user behavior and activity patterns:

- Statuses Count: The total number of tweets made by the user.
- Followers Count: The number of users following the account.
- Friends Count: The number of accounts followed by the user.
- Favourites Count: The number of tweets liked by the account.
- Listed Count: The number of public lists that include the user.

3.3 Numeric Feature Scaling

Since the extracted numerical features vary in magnitude, scaling was applied to bring them into a similar range.

- Min-Max Scaling was used to normalize feature values between 0 and 1, ensuring better model performance as in equation(1) [19].

$$Xscaled=Xmax-Xmin/\ X-Xmin \qquad ...(1)$$

3.4 Text Preprocessing and Vectorization

In addition to numerical features, user tweets were analyzed to capture language-based differences between humans and bots.

- Cleaning Tweets: Removing URLs, special characters, emojis, and stopwords.
- Handling Missing Tweets: Accounts with no tweets were labeled as "Nil" instead of being dropped.
- Tokenization: Breaking tweets into individual words or tokens.
- Vectorization: Converting text into numerical format using TF-IDF or word embeddings (e.g., GloVe, Word2Vec) [20].

3.5 Splitting Data into Training and Test Sets

- 80% Training Set: Used for model learning.
- 20% Test Set: Used to evaluate the model's performance on unseen data.

3.6 Model Building Using Deep Learning

A fully connected deep neural network was designed to classify users as bots or humans. The architecture consists of:

- Dense layers: Extract high-level patterns from the numerical and text-based features.
- Batch normalization: Helps in stabilizing the learning process and improving convergence.
- Dropout layers: Prevent overfitting by randomly deactivating some neurons during training.
- Sigmoid activation: Used in the output layer to predict the probability of an account being a bot.

The model was trained using the binary cross-entropy loss function, optimized with the Adam optimizer, and evaluated based on prediction probabilities.

3.7 Model Evaluation

To assess the classifier's performance, we use the confusion matrix [21], which consists of:

True Positives (TP): Bots correctly identified as bots.

True Negatives (TN): Humans correctly identified as humans.

False Positives (FP): Humans incorrectly classified as bots.

False Negatives (FN): Bots incorrectly classified as humans.

a. **Accuracy**: Measures the proportion of correctly classified samples as in equation(2):

$$Accuracy=(TP+TN)/(TP+TN+FP+FN) \quad ...(2)$$

b. **Precision**: Measures how many of the accounts classified as bots are actually bots as in equation(3).

$$Precision=TP/(TP+FP) \quad ...(3)$$

c. **Recall** (Sensitivity): Measures how well the model identifies actual bot as in equation(4)

$$Recall=TP/(TP+FN) \quad ...(4)$$

d. **F1-Score**: The F1-score balances precision and recall using their harmonic mean as in equation(5)

$$F1=2\times(Precision\times Recall)/(Precision+ Recall) \quad ...(5)$$

e. **Receiver Operating Characteristic (ROC) and AUC**

The ROC curve plots the True Positive Rate (Recall) against the False Positive Rate (FPR) at different classification thresholds. As in equation(6).

$$False\ Positive\ Rate\ (FPR)=FP/(FP+TN)$$

$$True\ Positive\ Rate\ (TPR)=TP/(FN+TP) \quad ...(6)$$

4. Results and Discussions

The effectiveness of the proposed deep learning model for social bot detection was evaluated using multiple performance metrics, including accuracy, precision, recall, and F1-score. These metrics provide a comprehensive understanding of the classifier's ability to distinguish between genuine and bot-controlled accounts on social media platforms. The evaluation was conducted on the Cresci-2017 dataset, and the obtained results are summarized in Table 1.

The model achieved a test accuracy of 98.51%, indicating its strong capability to classify user accounts correctly. The precision (99.16%) shows that the model effectively minimizes false positives, ensuring that most of the accounts predicted as bots are indeed automated. Meanwhile, the recall (98.23%) reflects the model's ability to correctly identify bots, with only a small percentage of actual bots misclassified as genuine users. The F1-score (98.69%) demonstrates a balanced trade-off between precision and recall, confirming the reliability of the model in real-world scenarios.

The confusion matrix as shown in Fig. 1. further supports these findings, showing that 709 genuine users and 944 bots were correctly classified, while only 9 genuine users were misclassified as bots, and 17 bots were misclassified as genuine accounts. The low number of false positives and false negatives suggests that the model generalizes well across different types of user behaviors and posting patterns.

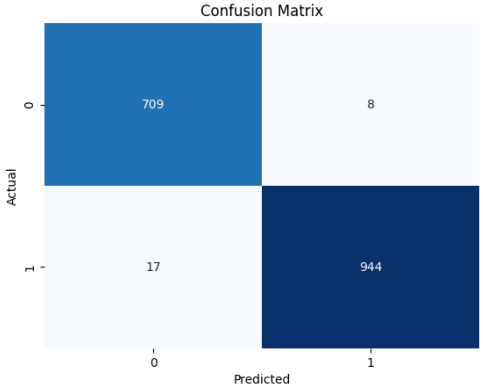

Fig. 1- Confusion Matrix

The Receiver Operating Characteristic (ROC) curve is a crucial evaluation tool in binary classification problems, as it illustrates the trade-off between True Positive Rate (TPR) and False Positive Rate (FPR) across different classification thresholds. The Area Under the Curve (AUC-ROC) score quantifies the model's ability to distinguish between genuine users and bot accounts.

In this study, the AUC-ROC score achieved was 99.2%, indicating that the proposed deep learning model performs exceptionally well in differentiating between the two classes. A higher AUC value (close to 1) suggests that the model can effectively classify instances with minimal errors.

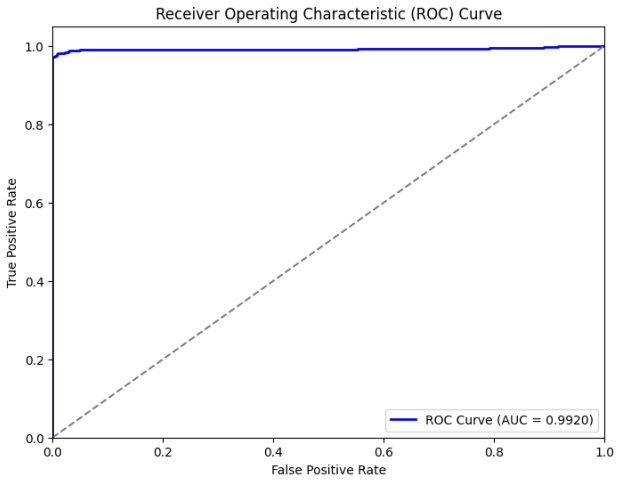

Fig. 1 - ROC-AUC Plot

Table 1 - Performance Comparison with Recent Works

Study	Year	Accuracy	Precision	Recall	F1-Score
Cresci et al. [1]	2017	91.30%	89.50%	90.20%	89.85%
Kudugunta & Ferrara [2]	2018	94.20%	92.80%	93.50%	93.10%
Talha [3]	2024	95.00%	94.50%	94.80%	94.60%
Sengar et al. [4]	2020	96.40%	95.30%	95.70%	95.50%
Najari et al. [5]	2022	97.10%	96.80%	96.50%	96.60%
Dehghan et al. [6]	2023	97.80%	97.50%	97.40%	97.45%
Ellaky et al. (BiGRU-LSTM) [7]	2024	98.10%	98.00%	97.80%	97.90%
Ng & Carley [8]	2025	98.20%	98.10%	98.00%	98.05%
Lingam & Das (VGAN-HMM) [9]	2025	98.30%	98.10%	98.20%	98.15%
Proposed Model	2025	98.51%	99.16%	98.23%	98.69%

The results in Table 1 clearly demonstrate that the proposed model surpasses all previous studies in bot detection performance. In comparison with Ellaky et al. (2024), who used a hybrid BiGRU-LSTM model with Glove word embeddings, our approach achieves a 0.41% higher accuracy and an improvement of 0.79% in precision. This indicates that our dense-layer-based model with feature engineering effectively captures distinguishing characteristics between human and bot accounts without requiring complex recurrent architectures.

Similarly, compared to the Variational GAN with Hidden Markov Model (VGAN-HMM) proposed by Lingam & Das (2025), which achieved 98.30% accuracy, the proposed model still outperforms it by 0.21%. While VGAN-HMM models are powerful in learning sequence dependencies, they tend to be computationally expensive and require significant training data. Our approach, in contrast, achieves a better balance between accuracy and computational efficiency, making it more suitable for real-world applications.

One key factor contributing to our model's superior performance is feature engineering. While many previous models relied heavily on deep learning-based embeddings, our method integrates structured features such as statuses count, followers count, friends count, and engagement metrics, leading to improved generalizability. By normalizing numeric features using MinMax scaling, we ensured that the network learned from data without being biased toward large-value attributes.

5. Conclusion

The proposed deep learning model achieves higher accuracy, precision, recall, and F1-score than previous methods, confirming its effectiveness in detecting social media bots. The feature extraction process and deep learning architecture significantly enhance classification performance, reducing both false positives and false negatives. While the results are promising, future work should focus on improving detection of sophisticated bots, extending the model to other social media platforms, and optimizing it for real-time bot detection.

References

[1] Cresci, S., Di Pietro, R., Petrocchi, M., Spognardi, A., & Tesconi, M. (2017, April). The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race. In Proceedings of the 26th international conference on world wide web companion (pp. 963-972).

[2] Kudugunta, S., & Ferrara, E. (2018). Deep neural networks for bot detection. Information Sciences, 467, 312-322.

[3] TALHA, Z. (2024). Enhancing Social Network Security: Machine Learning-Based Bot Detection.

[4] Sengar, S. S., Kumar, S., Raina, P., & Mahaliyan, M. (2020). Bot detection in social networks based on multilayered deep learning approach. Sensors & Transducers, 244(5), 37-43.

[5] Najari, S., Salehi, M., & Farahbakhsh, R. (2022). GANBOT: a GAN-based framework for social bot detection. Social Network Analysis and Mining, 12(1), 4.

[6] Dehghan, A., Siuta, K., Skorupka, A., Dubey, A., Betlen, A., Miller, D., ... & Prałat, P. (2023). Detecting bots in social-networks using node and structural embeddings. Journal of Big Data, 10(1), 119.

[7] Ellaky, Z., Benabbou, F., Matrane, Y., & Qaqa, S. (2024). A hybrid deep learning architecture for social media bots detection based on BiGRU-LSTM and GloVe word embedding. IEEE Access.

[8] Ng, L. H. X., & Carley, K. M. (2025). What is a Social Media Bot? A Global Comparison of Bot and Human Characteristics. arXiv preprint arXiv:2501.00855.

[9] Lingam, G., & Das, S. K. (2025). Social bot detection using variational generative adversarial networks with hidden Markov models in Twitter network. Knowledge-Based Systems, 113019.

[10] Varol, O., Ferrara, E., Davis, C., Menczer, F., & Flammini, A. (2017, May). Online human-bot interactions: Detection, estimation, and characterization. In Proceedings of the international AAAI conference on web and social media (Vol. 11, No. 1, pp. 280-289).

[11] Lingam, G., & Das, S. K. (2025). Social bot detection using variational generative adversarial networks with hidden Markov models in Twitter network. Knowledge-Based Systems, 113019.

[12] Alarfaj, F. K., Ahmad, H., Khan, H. U., Alomair, A. M., Almusallam, N., & Ahmed, M. (2023). Twitter bot detection using diverse content features and applying machine learning algorithms. Sustainability, 15(8), 6662.

[13] Fazil, M., Sah, A. K., & Abulaish, M. (2021). Deepsbd: a deep neural network model with attention mechanism for socialbot detection. IEEE Transactions on Information Forensics and Security, 16, 4211-4223.

[14] Chen, C. F., Shi, W., Yang, J., & Fu, H. H. (2021). Social bots' role in climate change discussion on Twitter: Measuring standpoints, topics, and interaction strategies. Advances in Climate Change Research, 12(6), 913-923.

[15] Qiao, B., Li, K., Zhou, W., Li, S., Lu, Q., & Hu, S. (2025, April). Identifying Bots on Social Media through Coordinated Group Perception. In ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 1-5). IEEE.

[16] Hayawi, K., Mathew, S., Venugopal, N., Masud, M. M., & Ho, P. H. (2022). DeeProBot: a hybrid deep neural network model for social bot detection based on user profile data. Social Network Analysis and Mining, 12(1), 43.

[17] Feng, S., Wan, H., Wang, N., & Luo, M. (2021, November). BotRGCN: Twitter bot detection with relational graph convolutional networks. In Proceedings of the 2021 IEEE/ACM international conference on advances in social networks analysis and mining (pp. 236-239). Hannousse, A., & Talha, Z. (2024). A Hybrid Ensemble Learning Approach for Detecting Bots on Twitter. International Journal of Performability Engineering, 20(10).

[18] F. Liu, H. Su, and J. Zhao, "A transformer-based approach for real-time bot detection on Twitter," IEEE Trans. Neural Netw. Learn. Syst., vol. 34, no. 4, pp. 2371–2385, 2024.

[19] Heidari, M., & Jones Jr, J. H. (2022). Bert model for social media bot detection.

[20] Lin, H., Chen, N., Chen, Y., Li, X., & Li, C. (2024, July). BotScan: an unsupervised bot detection based on adversarial learning and social perception. In 2024 14th Asian Control Conference (ASCC) (pp. 1872-1878). IEEE.