A Comparative Study of Learning Paradigms for Social Media Bot Detection

Samuel Adeniyi, Kerem Dogan, Luis Herrera
Artificial Intelligence (CSCI-7130)

Abstract

As of today, social media is one of the most common forms of communication used [1]. According to Statista, a global data and business intelligence platform, there are 5.56 billion internet users, and 5.24 billion of them are social media users [2]. These platforms have become a place where users can share information in real time and directly from the source; faster than any traditional media. However, there are malicious actors who want to take advantage of this fast spread of information to spread misinformation. They achieve this through the use of Social Bots, or bots. Software Agents that communicate automatically with social sites to manipulate engagement, push bias data, and spread misinformation. It would be impossible to manually review every account to determine if it is a bot or a human, so the application of Artificial Intelligence (AI) is used to automate the process. Recent developments in social media bot detection with AI have adopted deep learning algorithms such as Recurrent Neural Networks (RNN) and transformer-based models Bidirectional Encoder Representations from Transformers (BERT) [3]. However Social bots also use AI as well, creating a digital arms race between bots and the detectors. Which will lead to companies spending more money and resources to solve this issue. Our paper proposes that Machine learning, an old AI algorithm, still has its applications in bot detection and is a better alternative than the current method. The goal of this paper is to perform a comparative analysis of Machine learning supervised learning approach algorithms and Deep learning unsupervised learning approach algorithms [19]. Measuring their accuracy, precision, and F-1 when determining if an account is human or a bot with a dataset from Twitter. We are examining which of the models we have developed using different approaches and different techniques is more effective in solving this problem.

Teammates Roles and Contributions

All three team members had focused on implementing and analyzing an algorithm; Samuel worked on the Supervised algorithms RNN and Random Forest, Kerem also worked on Supervised algorithm, such Support Vector Machine (linear and RBF), a Self-Supervised Learning approach, and the model BERT, and Luis implemented the Unsupervised algorithms K-means and Isolation Forest. In the end, we're able to create a comprehensive report comparing each result of each algorithm. The end goal is to deliver a performance comparison report with accuracy, precision, recall, and F1-score.

1. Introduction

In today's world, the most popular form of communication is social media. Users are able to communicate with others from the other side of the world. Statista, a global data and business intelligence platform, states that there are 5.24 billion social media users online right now in 2024 [2]. Platforms such as Twitter (now known as X), Instagram, Facebook, and Tiktok have become part of the daily lives of many users. These sites allow for users to create and upload contents that can be shared with other users with no cost. However some people who share this information have the desire to purposely share the incorrect information to spread an agenda or to cause discourse. Since human interaction is a key part of social media platforms, it is assumed that other users will come and correct this information and stop the spread of this attempt to sow discourse. What about a case where the poster of misinformation is not even humans and it's being prompted up by other non-human users?

The modern definition of a Social bot is a computer algorithm that produces content and interacts with humans on social media [17]. It should be noted that social bots are not inherently malicious on social media. Accounts that post up to date news, weather, or entertainment updates use bots. They aggregate content from split sites and summarize it into posts. However, there lies the biggest potential danger in using social bots; the spread of information.Especially by those who wish to share misinformation across the site. Bots that appear posing as human accounts are the main sources on how these unverified stories can spread at exponential rates. Since these bots are programmed to mimic human behavior, they are incentivized to share this information to everyone. Social bots attract other Social bots, leading to them following each other. While there is not a confirmed hard number of the number of bots online but there have been trends noticed in the spikes in the use of them. Around election season there is a noticeable uptick in the bot traffic for obvious reasons [5]. As years go on, social bots increasingly become more sophisticated and it becomes more difficult to determine what is a human user and bot. This has led to a digital arms race to detect bots and remove bots from their sites. To achieve this goal, multiple forms of social bot detection methods have been applied, with the most successful ones incorporating artificial intelligence.

The core of bot detection is to use the data of social media accounts to determine if it simulates real human behavior. Since bots tend to be rapidly created on the same day and appear to be exceedingly recent compared to human operated accounts. When organizations need to find these non-human accounts they use three approaches; bot detection systems based on social network information, systems based on crowdsourcing and leveraging human intelligence; and, machine-learning methods based on the identification of highly revealing features that discriminate between bots and humans [17]. The most up to date technique is using the machine learning approach due to its advantage of focusing on behavioral patterns that can train algorithms to help them label accounts as either human-like and bot-like. While this approach to bot detection has been proven to be highly effective, social bots continue to change and grow which may lead to the machine learning approach becoming outdated.

2. Motivation

These social bots are not just a nuisance to software developers but to everyone who uses the social platforms. Bots adapt quickly, generate massive amounts of data, and often mimic human patterns, making manual detection ineffective. This is why AI algorithms are used, but which algorithm is the best one to use and which is most effortlessly able to be implemented? Deep learning is a subfield of machine learning known for its predictive accuracy [4]. Although deep learning algorithms provide higher accuracy in determining which accounts are bots compared to machine learning, machine learning models can be efficient and have easier outputs to intercept. Companies spend up to thousands and millions of dollars to handle cybersecurity threats consequently there will be thought on which AI model should be used to automate the process of bot detection. Most of the existing studies typically focus on a single modeling technique. They are providing limited insight into broader methodological differences. The main objective of this study is to compare the learning paradigms in machine learning and deep learning algorithms to see which model is stronger at detection of non-human accounts. The argument that we are striving to reach is to show that machine learning is not an obsolete method of bot detection and still has its uses in the current day. Since machine learning and deep learning are part of the same model of AI, the learning approach of each algorithm will be used. The two learning algorithms that will be analyzed are the Supervised learning approach and Unsupervised learning approach. The algorithms studied in the Supervised learning approach will be RNN, random forest, Support Vector Machine (SVM). The Unsupervised learning approach algorithms are Bidirectional Encoder Representations from Transformers (BERT), K-means, and isolation forest. After comparing each algorithm, a conclusion will be reached on which learning approach, in turn which AI model, is superior in bot detection and the current state of Machine learning in the field.

3. Literature Review

Supervised learning is a machine learning algorithm that uses labeled inputted data to make predictions based on patterns within the dataset.The authors of *Supervised Machine Learning Bot Detection Techniques to Identify Social Twitter Bots,* P. George, S. Payne, and N. Proferes use supervised learning techniques to identify bots on Twitter (known as X) and the features that accompany them. The data in the dataset are labeled as social spambots, traditional spambots, fake followers, and real users. The two largest areas in the dataset are real users first and social spambots second in the total number of tweets. Before the identification of bots takes place, the article states the three basic areas for analysis; profile, account activity, and text mining. The supervised learning model algorithm used in this article is Support Vector Machine (SVM). The reason why the authors chose this supervised machine learning algorithm is because, “the efficacy of the model is evaluated by the misclassification rate (error rate) and the true positive rate…A low misclassification rate means we are not misidentifying accounts owned by real people as bots” [6]. In the end, the results of using supervised learning techniques had a total accuracy rate 97.75% and 2.25% across all the types of bots listed [6]. This article provides a great insight on how accurate and precise supervised machine learning algorithms can be in bot detection. A breakdown on how the dataset will be broken down into categories and the labels used to create the binary classification used by the algorithm. The biggest strength of the article was labeling the different types of bots in the dataset and the areas that the algorithm will analyze to make a decision. The biggest limitation of the article is also the dataset itself. The dataset is from 2017, making it outdated compared to current data. The authors bring up the ethical issues that come with collecting data since real human accounts are also required to create a baseline. Making the case that outdated data is necessary to avoid using actual user data that may lead to an innocent user being doxxed. In conclusion, this article provides an in-depth showcase on supervised machine learning algorithms, especially Support Vector Machine, to demonstrate that they are extremely accurate in the application of bot detection on social media.

[20] Heidari et al. (2021) examined how sentiment-based features affect the success of machine learning models used to detect bots on social media. Considering bots' potential to create false trends and public opinion, they incorporated psychological and social factors into feature engineering using confirmation bias and the backfire effect. Using the Cresci 2017 dataset, which includes real people and different types of bots on Twitter, the study extracted polarity scores from tweets for each account using TextBlob and classified these values. The researchers observed that bots gave extreme and unidirectional emotional responses within a limited topic area, while humans gave more diverse emotional responses, which they attributed to confirmation bias and the backfire effect. The study employed Random Forest, SVM (SVC and SVR), Logistic Regression, and FFNN, testing the models with both the original metadata features and the addition of new sentiment features. The results show that accuracy, F1 and MCC metrics increase as new emotions are added, and Random Forest (92.3% accuracy) and SVC (91.4% accuracy) have the highest performance, while Logistic Regression (87.4% accuracy) and MCC (88.7% accuracy) are relatively weak. In their study, [21] Cai et al. (2017) propose a deep learning-based model called BeDM, approaching bot detection from a more

behavior-focused perspective. While most previous studies rely on hand-crafted features, this work stands out for its attempt to model user behavior and textual content together. The researchers treat a user's tweet history as a temporal text sequence and learn its representation using a CNN-LSTM framework. Additionally, behavioral signals such as posting intervals and retweet-to-tweet ratios, along with DeepWalk-based network embeddings, are incorporated into the model. This combined approach appears to produce a more robust representation compared to models that use content alone. Experiments report that BeDM outperforms classical methods like Boosting, BoostOR, and Stweeler in terms of F1 score. The study's main contribution is demonstrating that combining behavioral cues with content truly makes a significant difference in bot detection. It's noteworthy that behavior complements patterns that text alone cannot capture. [22] Mohammad et al. (2019) explored the possibility of determining whether a user is a bot based on a single tweet, leaving aside traditional user profile features for bot detection and gaining a more nuanced perspective. The study showed that a CNN model learning solely from tweet content achieves higher accuracy than a traditional ANN approach that uses profile-based features. A very high accuracy of 98.71% was reported on the Cresci 2017 dataset. This result demonstrates how effective text-based models can be in capturing subtle differences in bots' linguistic patterns. The authors also noted that a hybrid model that combines behavior and content yields the best performance. This study offers practical value in reducing the need for heavy preprocessing or reliance on extensive user history in bot detection. [23] Feng et al. (2024) conducted a comprehensive study evaluating the role of LLMs in social bot detection, both in terms of opportunities and risks. The authors consider LLMs not only as new detection tools but also as potential threats that can be used in bot generation and evasion strategies. They present a “heterogeneous expert mix” framework that can handle diverse information types, such as text, metadata, and network structure, and show that instruction tuning with just 1,000 examples can produce robust results. In experiments conducted on Mistral-7B, LLaMA2-70B, and ChatGPT, some LLMs reportedly outperform existing methods by up to 9% in F1-score. Furthermore, the study also reveals that manipulation strategies driven by LLMs can significantly weaken existing detectors. Both the strong detection capacity and the high risk of misuse suggest that the field will become even more challenging in the future. The study is both timely and important because it clearly demonstrates the connection between bot detection and LLM security.

4. Methodology

In this section, we explain how we designed, trained, and evaluated the machine learning and deep learning-based models for social media bot detection. Our study consists of multiple stages such as dataset preparation, feature engineering, approach-specific model development and evaluation, and definition of performance metrics. Many of these stages have specific and distinctive details about the model and technique. In this study, where we examine the behavior and detection of bots in social media, with the aim of making the study as comprehensive as possible, we focus on three learning paradigms: supervised learning, self-supervised learning, and unsupervised learning. For the supervised learning approach, which trains a model using labeled data and is used in prediction and classification tasks, we use Recurrent Neural Network (RNN), Random Forest, Linear Support Vector Machine (SVM), and Radial Basis Function (RBF) SVM techniques; for the unsupervised learning approach, which is used to detect patterns and relationships using unlabeled data, we use K-Means and Isolation Forest techniques; for self-supervised learning approach which based on the logic of training a model by creating its own supervisory signals using unlabeled data, BERT technique was used. The aim of using these approaches and techniques was to answer the following research question:

- How do classical machine learning models, deep learning architectures, and transformer-based methods differ in their ability to detect bot-like behavioral patterns in social media data?

4.1 Dataset

The Twitter Human-Bots Dataset contains rich profile information to distinguish real and bot accounts and was used in the training and evaluation processes of our social media bot detection models. This structure, consisting of 23 attributes, encompasses both behavioral and metadata-based signals such as account creation date, follower and following counts, tweet frequency, account age, profile image usage, verified account information, and account biography. The account_type label was used as the target feature in our supervised learning models. The split column in the dataset facilitated the creation of training and test subdatasets.

4.2 Supervised Learning Approaches

4.2.1 Recurrent Neural Network (RNN)

RNN is a deep neural network, a machine learning model that has layers that have pattern-recognizing weights and biases from data to map inputs to outputs. It is also important to note that RNN can be either a supervised, the input data is labeled, or unsupervised, the input data is not labeled, learning approach. Since deep learning methods used in bot detection uses unsupervised learning, the RNN algorithm used in this project will be supervised [7]. What makes RNN stand out is its ability to make sequential predictions or conclusions based on sequential inputs. In social media bot detection, RNN is highly useful for finding the relationship in Temporal/Behavioral features [18]. However, RNN fails when it comes to collecting information from long sequences.

To prepare the data for the RNN model, the variable ‘account type’, the attribute that determines if an account is a bot or human, will be one of the labels encoded. Then we need to choose another attribute to see if there is any correlation in determining the account type, in this case the attribute of ‘lang’(language) was chosen. The process begins by making ‘lang’ our input feature and ‘account_type’ as our target variable. The ‘account_type’ is converted into a boolean variable, 0 for bot and 1 for human. The model creates 3 layers; an embedding layer that takes the sequence for each language, a (Long Short-Term Memory)LSTM layer used to capture trends within the sequence, and finally a Dense layer to create the output from the input.

The end results showed some low scores compared to the other models used in the paper, especially when it came to identifying bots. However RNN shows promising results when it comes to determining human accounts. The RNN model did showcase that the majority of bot and human accounts came from English speaking countries.

4.2.2 Random Forest

Random Forest is a method that uses an "ensemble" of many decision trees to make more accurate predictions for classification and regression tasks. One of the main advantages that Random Forest excels in is classification by converting boolean values. With this algorithm, we can find which variable from the dataset has the most correlation in human and bot accounts. This model, Random Forest, will use the features associated with the accounts, such as ‘geo_enabled’, ‘verified’, and ‘account_age_days’, to determine correlations between human and bot accounts. The ‘account_type’ attribute will be the target variable, known as the ‘correlation_df’ and the features ‘geo_enabled’, ‘verified’, ‘default_profile’, ‘account_age_day’, ‘favoritves_count’, ‘defualt_profile_image’, ‘followers_count’, ‘average_tweets_per_day’, ‘statuses_count’, and ‘friend_count’.

The results show that Random forest has a very positive average when in accuracy and precision. Compared to the other learning algorithms used, it has the highest score in precision (0.75), Recall (0.84), and F-1 score (0.79). Making it the most ideal algorithm for detecting bots on social networks. The major discovery from this algorithm is that the three features that have a strong correlation of an account being a bot are ‘geo_enabled’, ‘verified’ and ‘default_profile’.

4.2.3 Linear SVM

A Linear SVM is a powerful machine learning model that uses a straight hyperplane to separate classes by assuming the data can be separated linearly. It performs well on high-dimensional text data such as TF-IDF features like we have in our dataset.

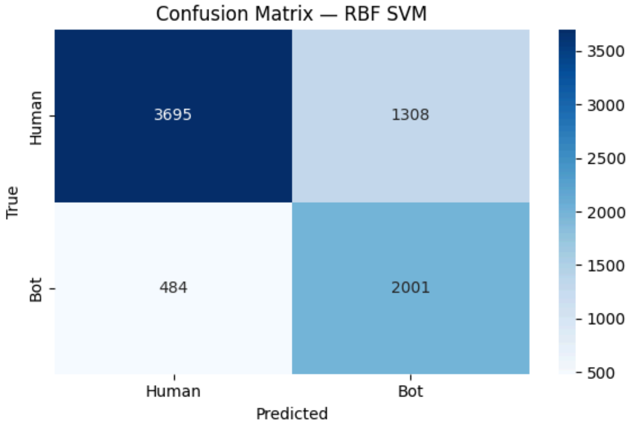
As part of preprocessing, all numerical features in the dataset were normalized, categorical features were transformed appropriately, and missing values were checked. The model was trained using only behavioral features such as tweet count, engagement rates, and follower count, as well as metadata. Using StandartScaler, the input features were standardized to a mean of 0 and a variance of 1, vectorization, and scaling were performed. The model was created using LinearSVC, and the optimal value for the regularization coefficient (C) was found. The model was then fitted to the training set of the dataset, and the training process was completed. Predictions were taken from the test set and the necessary metrics were calculated and reported. It achieved the highest accuracy rate among models to date, at 0.72. Due to its high recall and low precision values, it achieved an average F1 score of 0.65.

The SVM results indicate that metadata-based bot detection is highly effective, especially when non-linear kernels are used. The baseline Linear SVM achieved moderate performance but struggled with precision, indicating that it frequently misclassified human accounts as bots.

4.2.4 RBF SVM

An RBF SVM is a machine learning model for classification and regression that uses the RBF kernel to map data into a higher-dimensional space, allowing it to find non-linear decision boundaries by making them more flexible.

Preprocessing operations performed in Linear SVM were repeated. The RBF kernel was used, assuming bot behaviors that involve more complex patterns and cannot be linearly decomposed. This technique, called the 'kernel trick', allows linear models to handle nonlinear data by implicitly mapping it into a higher-dimensional space without explicitly computing the coordinates of the transformed data. The same normalization process is used here with StandardScaler, as the RBF kernel is sensitive to distance calculations. The model was fitted using the training set, and high-dimensional decision boundaries were created by learning the non-linear relationships between the profiles. Hyperparameter tuning was achieved by running GridSearchCV. Predictions were generated using the test set and reported. The model achieved a very high accuracy value of 0.77 and an F1 score of 0.70.



The confusion matrix shows that the optimized SVM model excels in identifying bots, exhibiting a high recall rate and a relatively low number of false negatives, though it does occasionally misclassify some human users as bots (false positives). These results indicate that traditional machine learning models, such as SVM, continue to be effective and dependable for tasks involving metadata-driven bot detection, delivering impressive performance with reduced computational demands and greater interpretability compared to contemporary deep learning methods.

4.3 Self-Supervised Learning Approach

Self-supervised learning (SSL) focuses on acquiring distinguishing features from data that is not labeled, without depending on labels provided by humans [24]. It is a Google AI language model that understands the context of words in text by reading them in both directions simultaneously.

4.3.1 BERT (Bidirectional Encoder Representations from Transformers)

The description section of the dataset, corresponding to the user's biography, was used as the sole textual input for the BERT model. In the preprocessing step, all biographies were meaningfully edited, and missing values were assigned an empty string. To ensure label construction, the account_type attribute was examined, and stratified sampling was applied to split the dataset into two parts: training and test, with an 80/20 ratio. The texts were converted to tensors using a BERT tokenizer, and then padding and truncation operations were applied. These tokenized samples were imported into HuggingFace and formatted for PyTorch, preparing them for the Trainer API. A classification layer was added to the pretrained model and fine-tuned as a two-class classifier. As a result of the evaluation tests, the model came very close to the RBF SVM model by achieving an accuracy value of 0.75 and showed an average harmonic mean value by achieving an F1 score of 0.64.

4.4 Unsupervised Learning Approach

4.4.1 K-Means

K-Means was used to group accounts into two clusters based on features like follower count, following count, account age, and posting activity. Because K-Means is unsupervised, it assigns clusters without knowing which accounts are bots or humans.

The results showed that the two groups were not clearly separated. The model had a recall of 0.91, meaning it identified most bot accounts. However, the precision was 0.50, so many human accounts were incorrectly grouped with bots. This suggests that using only numerical metadata is not enough for K-Means to reliably tell bots from human users.

4.4.2 Isolation Forest

Isolation Forest was used to identify unusual account behavior by identifying data points that stood out in the feature space. Ideally, bot accounts would act differently enough to be flagged as anomalies. However, the model grouped almost equal numbers of humans and bots in both clusters. Cluster 1 had 12,526 humans and 12,478 bots, while Cluster 0 had 12,456 humans and 12,540 bots. This resulted in an overall accuracy of 0.50. These results show that the features used, which were mostly numerical metadata, were not distinctive enough for Isolation Forest to tell the accounts apart. While the algorithm did spot some unusual patterns, it was not effective as the only method for finding bots in this dataset.

5. Results

Model	Learning Type	Accuracy	Precision	Recall	F1 Score
RBF SVM	Supervised (ML)	0.77	0.62	0.80	0.70
BERT	Self-supervised + Fine Tuned	0.75	0.61	0.69	0.64
Linear SVM	Supervised (ML)	0.72	0.56	0.79	0.65
Random Forest	Supervised (ML)	0.71	0.75	0.84	0.79
RNN	Supervised (DL)	0.66	0.49	0.50	0.48
Isolation Forest	Unsupervised	0.50	0.50	0.91	0.64
K-Means	Unsupervised	0.50	0.50	0.50	0.50

The table above provides a comparative analysis of the models' accuracy, generalization, and interpretability.

Among classical machine learning models, the RBF SVM demonstrated the strongest performance, detecting nonlinear patterns with impressively high accuracy. The Transformer-based BERT model demonstrated its prowess in contextual text naming by achieving results very close to and competitive with the RBF SVM. On the other hand, the Linear SVM, which was limited in analyzing nonlinear and complex relationships, performed relatively poorly compared to these models. Random Forest, which reveals attribute importance in an interpretable manner, delivered strong performance on numerical and structural metadata. While the RNN model has an advantage in processing sequential text data, its accuracy rate lags behind other models. While the Isolation Forest model achieves high success in its primary purpose of anomaly detection, its relatively high false positive rate undermines its use as a standalone model. Instead, it can be considered as a complementary and guaranteeing submodel to some main model. Finally, the K-Means algorithm generated mixed clusters and was unable to adequately distinguish bots from humans using only numerical features.

The project we're involved in involves many parameters, including dataset, preferred models, problem-solving approach, feature engineering, and accessible development environments. Based on the results we've obtained in this context, we can say that the RBF SVM model is the best solution to the existing social media bot detection problem.

6. Discussion

In this section, we will talk about the strengths and weaknesses of our work, its limitations, and how it can be improved in the future.

The most important strength of our study is its comprehensiveness, thanks to the large number of models developed and examined. Studies in the literature often focus on a single model, technique, or approach, and the goal is to refine and/or differentiate them. We believe a comprehensive study can provide a broader perspective and insight. Another strength is that we demonstrated that bot detection can be achieved with average accuracy without using labeled data. The self-supervised and unsupervised methods we used are examples of these.

One of the weaknesses in our project stems from the dataset we used. Our dataset contained mostly profile metadata and limited text, which restricted the ability of deep learning and unsupervised models to capture richer behavioral patterns. Using a more comprehensive dataset such as Cresci would likely improve model accuracy and generalizability. Another weakness is that the cluster structures and learning paradigms we used in certain models cause overlap problems. Consequently, we obtained slightly lower accuracy and F1 score values than expected in some of our models.

The one limitation we have is that the model’s performance is highly dependent on the quality of feature selection and scaling, as inappropriate feature representation or inconsistent normalization can substantially degrade the model’s ability to capture meaningful patterns and generalize to unseen data.

We believe that to achieve even better results in the future, we need a more comprehensive dataset and feature engineering process. Furthermore, developing reassurance mechanisms using hybrid models to obtain better metrics for our solutions could be another important future consideration.

7. Conclusion

Different learning models capture different characteristics of social media bot behavior, according to our comparative analysis. The best performance was shown by supervised models, especially RBF SVM, which successfully used behavioral and textual characteristics to differentiate between human and bot accounts. While BERT's transformer design gathered deeper contextual patterns and produced competitive accuracy and balanced precision-recall trade-offs, traditional machine learning models like Random Forest and SVM offered reliable and comprehensible baselines. Because numerical activity features alone did not form clearly separable clusters, unsupervised methods such as K-Means and Isolation Forest provided limited standalone classification capability. Nevertheless, they are still useful for anomaly detection and as auxiliary tools for identifying emerging bot behaviors.

References

[1] Adebayo, Olumide, and Elif Kongar. “Impact of Social Media Marketing on Business Performance: A Hybrid Performance Measurement Approach Using Data Analytics and Machine Learning.” *IEEE Engineering Management Review*, vol. 49, no. 1, 2021, pp. 1–1, https://doi.org/10.1109/emr.2021.3055036 .

[2] Ani Petrosyan. “*Number of Internet and Social Media Users Worldwide as of February 2025*.” Statista, 1 Apr. 2025, www.statista.com/statistics/617136/digital-population-worldwide/.

[3] Dukić, David, et al. “Are You Human? Detecting Bots on Twitter Using BERT.” *IEEE Xplore*, 1 Oct. 2020, ieeexplore.ieee.org/document/9260074.

[4] Ekundayo, Olufisayo S., and Absalom E. Ezugwu. “*Deep Learning: Historical Overview from Inception to Actualization, Models, Applications and Future Trends.*” *Applied Soft Computing*, vol. 181, Sept. 2025, p. 113378, https://doi.org/10.1016/j.asoc.2025.113378.

[5] Orabi, Mariam, et al. “Detection of Bots in Social Media: A Systematic Review.” Information Processing & Management, vol. 57, no. 4, 1 July 2020, p. 102250, www.sciencedirect.com/science/article/abs/pii/S0306457319313937, https://doi.org/10.1016/j.ipm.2020.102250.

[6] Phillip George Efthimion, et al. “Supervised Machine Learning Bot Detection Techniques to Identify Social Twitter Bots.” *SMU Scholar*, 2018, scholar.smu.edu/datasciencereview/vol1/iss2/5/.

[7] Ping, Heng, and Sujuan Qin. “A Social Bots Detection Model Based on Deep Learning Algorithm.” *IEEE Xplore*, 1 Oct. 2018, ieeexplore.ieee.org/document/8600029?arnumber=8600029. Accessed 19 Oct. 2022.

[18] Hayawi, Kadhim, et al. “Social Media Bot Detection with Deep Learning Methods: A Systematic Review.” Neural Computing and Applications, vol. 35, 6 Mar. 2023, https://doi.org/10.1007/s00521-023-08352-z.

[19] Gandhi, Rishabh, and Yanyan Li. Comparing Machine Learning and Deep Learning for IoT Botnet Detection. 1 Aug. 2021, https://doi.org/10.1109/smartcomp52413.2021.00053. Accessed 2 Feb. 2024.

[20] Heidari, M., Jones, J. H., Jr., & Uzuner, O. (2021). An empirical study of machine learning algorithms for social media bot detection. In 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS) (pp. 1-7). IEEE. https://doi.org/10.1109/IEMTRONICS52119.2021.9422605

[21] Cai, C., Li, L., & Zeng, D. (2017). Behavior enhanced deep bot detection in social media. In 2017 IEEE International Conference on Intelligence and Security Informatics (ISI) (pp. 128-130). IEEE. https://doi.org/10.1109/ISI.2017.8004887

[22] Mohammad, S., Khan, M. U. S., Ali, M., Liu, L., Shardlow, M., & Nawaz, R. (2019). Bot detection using a single post on social media. In 2019 Third World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4) (pp. 215-220). IEEE. https://www.researchgate.net/publication/337528436_Bot_detection_using_a_single_post_on_social_media

[23] Feng, S., Wan, H., Wang, N., Tan, Z., Luo, M., & Tsvetkov, Y. (2024). What does the bot say? Opportunities and risks of large language models in social media bot detection. arXiv preprint arXiv:2402.00371v2. https://arxiv.org/abs/2402.00371

[24] Gui, J., Chen, T., Zhang, J., Cao, Q., Sun, Z., Luo, H., & Tao, D. (2024). A survey on self-supervised learning: Algorithms, applications, and future trends. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 46*(12), 9052-9075. https://doi.org/10.1109/TPAMI.2024.3415112

Appendix

[8] Jones, Matthew. "*The History of Social Media: Social Networking Evolution!*" History Cooperative, 31 Oct. 2024, historycooperative.org/the-history-of-social-media/.

[9] "*Evolution of Social Media Algorithms: The Invisible Hand Guiding Our Online Experience | Reso*." Reso, 4 May 2025, resoinsights.com/insight/evolution-of-social-media-algorithms/.

[10] Ng, L.H.X., Carley, K.M. A global comparison of social media bot and human characteristics. *Sci Rep* 15, 10973 (2025). https://doi.org/10.1038/s41598-025-96372-1

[11] T.K., Balaji, et al. "*Machine Learning Algorithms for Social Media Analysis: A Survey*." Computer Science Review, vol. 40, no. 1, May 2021, https://doi.org/10.1016/j.cosrev.2021.100395.

[12] 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.* https://doi.org/10.1109/ACCESS.2024.3430859

[13] Darem, A. A. Alhashmi, M. H. Alanazi, A. F. Alanezi, Y. Said, L. A. Darem, and M. M. Hussain, "Cybersecurity in social networks: An ensemble model for Twitter bot detection," *International Journal of Advanced and Applied Sciences*, vol. 11, no. 11, pp. 130–141, 2024. [Online]. Available: https://doi.org/10.21833/ijaas.2024.11.014

[14] Davis, O. Varol, F. Ferrara, A. Flammini, and F. Menczer, "BotOrNot: A system to evaluate social bots," in *Proc. 25th Int. Conf. Companion World Wide Web*, 2016, pp. 273–274. [Online]. Available: https://doi.org/10.1145/2872518.2889302

[15] Ouni, F. Fkih, and M. N. Omri, "Bots and gender detection on Twitter using stylistic features," in *Advances in Computational Collective Intelligence*, C. Bădică, J. Treur, D. Benslimane, B. Hnatkowska, and M. Krótkiewicz, Eds. Cham, Switzerland: Springer, 2022, pp. 650–660. [Online]. Available: https://doi.org/10.1007/978-3-031-16210-7_53

[16] Wu, Wei, et al. "Bot Detection Using Unsupervised Machine Learning." *Microsystem Technologies*, vol. 24, no. 1, 31 Dec. 2016, pp. 209–217, https://doi.org/10.1007/s00542-016-3237-0.

[17] The Rise of Social Bots – Communications of the ACM. 1 July 2016, cacm.acm.org/research/the-rise-of-social-bots/.

[25] Liu, X., Zhang, F., Hou, Z., Mian, L., Wang, Z., Zhang, J., & Tang, J. (2023). Self-supervised learning: Generative or contrastive. *IEEE Transactions on Knowledge and Data Engineering, 35*(1), 857-872. https://doi.org/10.1109/TKDE.2021.3090866