**Journal of Computer Science and Technology Studies**
ISSN: 2709-104X
DOI: 10.32996/jcsts
Journal Homepage: www.al-kindipublisher.com/index.php/jcsts

JCSTS
AL-KINDI CENTER FOR RESEARCH AND DEVELOPMENT

**| RESEARCH ARTICLE**

# Detection Technology of Social Robot: Based on the Interpretation of Botometer Model

**Jiawen Tian[1] ✉ Yiting Huang[2] and Dingyuan Zhang[3]**
[1]*School of Computer Science Engineering, University of New South Wales, Sydney, Australia*
[2]*School of Journalism and Communication, Nanjing Normal University, Nanjing, China*
[3]*School of Communications and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing, China*
**Corresponding Author:** Jiawen Tian, **E-mail**: govern.consultation@gmail.com

**| ABSTRACT**
In the era of Web 2.0, social media have been a significant place for democratic conversation about social or political issues. While in many major public events like the Russia-Ukraine war or U.S. Presidential election, enormous social bots were found on Twitter and Facebook, putting forward public opinion warfare. By creating the illusion of grassroots support for a certain opinion, this kind of artificial intelligence can be exploited to spread misinformation, change the public perception of political entities or even promote terrorist propaganda. As a result of that, exploiting detection tools has been a great concern since social bots were born. In this article, we focused on Botometer, a publicly available detection tool, to further explain the AI technologies used in identifying artificial accounts. By analyzing its database and combing the previous literature, we explained the model from the aspect of data augmentation, feature engineering, account characterization, and Ensemble of Specialized Classifier (ESC). Considering the consistent evolution of social bots, we propose several optimization suggestions and three other techniques or models to improve the accuracy of social bots detection.

**| KEYWORDS**

Social Bots; Twitter; Botometer; Data Augmentation; Feature engineering

**| ARTICLE DOI: INFORMATION**

**ACCEPTED:** 30 August 2022 **PUBLISHED:** 30 August 2022 **DOI:** 10.32996/jcsts.2022.4.2.6

## 1. Introduction

Currently, social media is extremely widely used in the communication and democratic discussion of political and social issues. However, in several public events such as the US election, the Brexit referendum, the Hong Kong amendment fiasco, and reports of the new crown epidemic, a large number of social media users who disseminated information were proven not to be human but social bots controlled by automated programs (Shi & Chen, 2020). In their study of the 2016 US election, Bessi and Ferrara (2016) found that about 14% of topic-related tweets and accounts were automated programs. In a study by Howard et al. (2018), 36.1% of pro-Trump tweets and 23.5% of pro-Hillary tweets were driven by social bots. The social media ecology is taking on a new character of "people + social bots (Zhang et al., 2019). The widespread use of social bots in the political, economic, and social spheres has not only promoted the development of intelligent communication but also provided a breeding ground for the dissemination of disinformation and malicious communication, which may lead to the imbalance of the communication ecology and the diffusion of disinformation (Zheng & Fan, 2020).

## 2. Literature Review

### *2.1 The Mechanism of Social Bots*

A social bot is a computer algorithm that automatically produces content and interacts with humans on social media (Liu et al., 2017). According to Shiwen and Chen Changfeng, social bots mimic the behavior of human users in posting messages and

following other users in order to guide public opinion and manipulate public perception (Shi & Chen, 2020). Norah Abokhodair explained the social bot as an automated social actor, which acts like how a person behaves in a social space, posing as a person or organization to promote a particular ideology to create a false sense of consensus (Guo & Zhao, 2022). By mimicking human behavior, bots can go unnoticed and improve their chances of influencing the social graph.

As we enter the web 2.0 era, the human-content relationship is deepened into a human-person relationship, with 'personal portals' becoming the dominant form of information distribution in social media. Each node in a social network plays multiple roles as a producer, distributor, and receiver, and information flows along social relationships. In this environment, social bots need to improve the credibility of their accounts and build social networks to facilitate the dissemination of information. Previous research has shown that social bots "interact socially" like human users, engaging in interactions through "likes" to gain emotional support and social capital (Shi & Wang, 2022). By actively following human accounts, they "interconnect" with human users, thus gradually building up a social network with other users (Lu et al., 2021).

Social bots post polarized and emotional content, focusing on the mass publication of relevant information on a topic within a short time (Chen & Yuan, 2021). In this way, they influence people's perception of the importance of a topic and achieve agenda-setting. On this basis, the rhetoric of social bots guides the audience's judgment of a certain event (Shi & Wang, 2022). Human users are prone to the illusion that this information or opinion is the popular opinion due to the herd mentality. As a result, they are manipulated by social bots to set the value stance. Zheng Chenyu and Fan Hong analyze the mechanisms by which social bots drive social diffusion from the perspective of structural functionalism. For the dissemination of simple information content, social bots tend to achieve "simple contagion" through "Small-World" networks with many weak connections. For "complex contagion", where attitudes, opinions, and behaviors are intermingled, social bots tend to build dynamic structural types similar to cluster networks to facilitate social diffusion.

Other scholars have found that the spread of suspected social bot accounts is more pronounced in the first few seconds after an article is first published on Twitter. In the early stages of dissemination, social bots increase the exposure of the message, improving the chances of the content going 'viral'. They generate massive amounts of messages, facilitating astroturfing or Twitter bombs with the effect of suppressing opposing candidates, polarizing controversy, and diverting audience attention (Zheng & Fan, 2020). As such, social bots generally employ an opinion amplification strategy rather than a guidance strategy; they are not opinion leaders in communication and are closer to the boisterous masses (Lu et al., 2021).

### *2.2 Detection of Social Bots*

In 2012, social media was born, and the defense and identification of new botnets began to attract academic attention. In 2017, AI became a hot topic in the field of social bot identification and detection. Previous literature shows that the use of AI is common for automated account detection in computer science. On the one hand, techniques such as image recognition and convolutional neural networks have been used to enhance bot account compulsions; while on the other hand, AI techniques such as machine learning and natural language processing have been actively applied to optimize bot account detection models (Zheng & Han, 2021).

Using Twitter as an example, Ferrara and other scholars summarized three mainstream techniques for social bot recognition, namely: social network information, or graph-based method; human intelligence detection system based on crowdsourcing-based sourcing; and machine learning method based on feature engineering (Ferrara et al., 2016).

Among these, the machine learning method is the dominant detection method for social robots. It typically applies machine learning algorithms to the accounts to be detected to determine whether they are social bots or humans. Social bots are detected by extracting simple user characteristics using Bayesian models, K-nearest neighbor models, C5 decision trees, as well as deep learning. Social network information, or graph-based methods, are used to characterize the different social associations between social bots and normal users. It transforms the asocial bot detection into a graph node classification problem and then uses graph mining algorithms to distinguish between human accounts and social bot accounts. In human intelligence detection system based on crowdsourcing-based sourcing, crowdsourcing bots split the social bot detection task into smaller tasks that are distributed to a large number of volunteers or paid humans. Compared to robotic detection, humans are far better able to assess conversational nuances and observe emerging patterns and anomalies than machines. However, the high costs of a multi-user platform should be taken into account while applying this method.

Though bot detection has become a hot field for academic study, little research has been done to explain the deeper principles of social bot recognition technology, making the technology a 'black box' to users. The evolution of social bots towards stealthier and human-like detection technologies has placed greater demands on the improvements of detection technologies. Therefore, based on the Botometer, a web-based program that identifies Twitter accounts by machine learning, this paper explains its core

principles and important features of social bots detection. Combined with new features of contemporary social robots and other detection techniques, we also give practical advice and propose optimized techniques to provide a reference for the improvement of detection techniques.

## 3. Methodology

In this paper, we mainly use documentary analysis and literature reviews to conduct our research. In particular, we analyze the sequential processes in Botometer with technique documents such as API manuals and tutorials. By reviewing papers and conferences available in public libraries and institutions, we investigate and interpret the additional datasets that Botometer has been fed into and categorize the further features stemming from these augmented data. Moreover, after pointing out that the main issue of the current Botometer is over data dependency, we cite various state-of-the-art techniques and models to evidence that there are multitudinous methods that can be utilized for reference and transfer, which eventually can improve the existing model.

## 4. Botometer Evaluation

Botometer, formerly called BotOrNot, was developed by Indiana University. It is a machine-learning algorithm that uses machine learning to classify Twitter accounts as bots or humans. Twitter users can log in and check the Botometer scores for their followers and other Twitter users. Based on tens of thousands of labeled examples, it can check the activity of Twitter accounts and rate how likely a Twitter account is to be a bot. In recent years, the Botometer platform has become the primary tool for social bot account determination.

Currently, Botometer has been able to achieve a 95% recognition rate and is popular in the academic community for its powerful and comprehensive nature (Shi & Chen, 2020).

### *4.1 Data Augmentation*

The performance of supervised machine learning models is greatly influenced by the training data. Social robotics is evolving rapidly, and even the most advanced algorithms can fail due to outdated training datasets. To solve the problem of generalization, Botometer introduced larger and more complex datasets for training.

- Time zone: An account will be suspected if its profile shows the US Eastern Standard time zone and most of its followers appear to be in the Moscow Standard Time zone.
- Language metadata: Abnormal patterns in language use and audience language preferences can be revealed by the low proportion of neighbors (friends, followers) who share the same language as the target account.
- Device metadata: We capture the type of device and platform used to post tweets, as well as the entropy across platforms.
- Content deletion mode: Highly active accounts frequently create and delete content to hijack users' attention without revealing much about their excessive Posting behavior. We incorporate features to estimate content production rates based on account creation dates, metadata on total tweets and recent in-data capacity, and time between events for recent posts -- mismatches between these statistics can serve as proxies for content removal.

The Botometer not only ensures the dimensionality of the analysis but also normalizes the dataset. The common content of tweets is informal text, which may consist of text, emoticons, pictures, symbols, and web links. And this non-standard text will bring the block to text content recognition. In order to remove text noise, better identify the real text meaning of the content text.

At the same time, the classification model is updated to continuously extract and screen features to distinguish between human behavior and increasingly complex robot behavior using newly acquired data as well as feedback collected from users. Botometer's rich and perfect database expansion and preprocessing also provide richer samples and more multidimensional selection for the next step of feature selection, account characterization, and clustering (Fazil & Abulaish, 2017).

Finally, the performance on training datasets and robustness on unknown datasets are measured by cross-validation, and the robot detection model is retrained by new data and feature engineering. As a result, the Botometer improved the accuracy of the model and also enriched the list of functions used, adding new functions designed to capture the robot used in the information operations (Sayyadiharikandeh et al., 2020).

Table 1 The training data used by Botometer. (CNetS stands for the Center for Complex Networks and Systems Research at Indiana University)

| Dataset name | #Bots | #Human | Notes |
|---|---|---|---|
| **caverlee** | 22,179 | 19,276 | Honeypot-lured bots and sample human accounts (Lee et al., 2011) |
| **varol-icwsm** | 826 | 1,747 | Manually labeled bots and humans sampled by Botometer score deciles (Zheng & Fan, 2020) |
| **cresci-17** | 10,894 | 3,474 | Span bots and normal humans (Kloumann et al., 2012) |
| **pornbots** | 21,963 | 0 | Pornbots shared by Andy Patel (github.com/r0zetta/pronbot2) |
| **celebrity** | 0 | 5,970 | Celebrity accounts collected by CNetS team |
| **vendor-purchased** | 1,088 | 0 | Fake followers purchased by CNetS team |
| **Botometer-feedback** | 143 | 386 | Botometer feedback accounts manually labeled by author K.-C.Y. |
| **Political-bots** | 62 | 0 | Automated political accounts run by @rzazula (now suspended), shared by @josh_emerson on Twitter |
| **Total** | 57,155 | 30,853 | All datasets available at Botometer.iuni.iu.edu/bot-repository |

Source: Sayyadiharikandeh et al. (2020).

### *4.2 Feature Engineering*

Botometer applies random forest, one of the supervised learning models, to extract over 1000 features, which can be categorized into six classes: user profile, friends, network, temporal, content and language, and sentiment.

Random Forest is one of the ensemble methods, choosing the decision tree as the basic model and Gini Impurity as the dataset division principle. Decision Tree relies on the principle of maximum Information Gain, and it segments the dataset based on chosen features until it reaches a stop criterion such as minimum leaves or sample impurity. The entropy can be calculated as below:

$$Entropy(S, D) = \sum_{i=1}^{m} \frac{|S_i|}{|S|} Entropy(S_i)$$

$$IG(S, D) = Entropy(S) - Entropy(S|D)$$

The ensemble methods of machine learning can be classified into sequential ensembles, such as AdaBoost and Generalized Boosted Regression (GBM), and parallel ensembles like bagging or bootstrap aggregating. As shown in *Algorithm 1*, each data in dataset $D$ has an input $x$ and an output $y$, each weak model ζ predict $h(x)$, and for input $x$ the prediction result is the average, for regression, or voting result, for classification, of all weak model predictions. The training samples used for each weak model $D_n$ were derived from a bootstrap of the original dataset $D$.

**Algorithm 1:** Bagging

**Input**

$D$ Original Dataset $= \{(x_1, y_1), (x_2, y_2), ..., (x_m, y_m)\}$

$\zeta$ Base learnig algorithm

$N$ Number of base learner

**Process**

for $n = 1, ..., N$

$h_n = \zeta(D_n)$

**Output**

$H(x) = \arg\max_{y \in Y} \sum_{n=1}^{N} h_n(x) = y$

Random forest adopts the CART tree, which is based on the Gini coefficient for feature selection. The CART tree acts as a binary tree, splitting each feature, selecting the feature point with the smallest Gini coefficient, and dividing the data into two subsets until the stop condition is met. Calculate the Gini Impurity* for each feature:

$$GI(p) = \sum_{k=1}^{K} p_k(1 - p_k) = 1 - \sum_{k=1}^{K} p_k^2$$

$$GI(D, C = c) = \frac{|D_1|}{|D|} GI(D_1) + \frac{|D_2|}{|D|} GI(D_2), D_1 = D \; and \; C = c; D_2 = D \; and \; C \neq c$$

*Gini Impurity represents the probability that a randomly selected sample in a sample set will be misclassified; the larger the Gini index, the greater the uncertainty. The smaller the Gini coefficient, the smaller the uncertainty, and the more thoroughly and cleaner the data segmentation

Stemming from the bagging technique, random forest chooses Classification and Regression Trees (CART) as a weak model and appends randomness on random features extraction, generally defaulting to, where is the total number of features, and makes the sample size consistent, defaulting to the sample size of the original data.

Random Forest is an adopted ensemble algorithm with CART that makes it can handle multi-types of data, linear/non-linear data, and discrete/continuous data, be robust to data noise and outliers. Moreover, sample randomness and feature randomization make the random forest uneasy to overfit. Due to the out-of-pocket data (OOB), an unbiased estimate of the true error can be obtained during model generation without loss of training data amount. All these finally result in competitive performance and relatively faster training/testing speed.

Features contained in the user profile include the number of friends, the number of fans, the number of original Twitter, and the user interface description, which contains the user interface features including username length, whether the user uses the default picture, background, account age, etc. The friendship feature manifests itself in users connecting through the follower-followee relationship; meanwhile, tweets spread between users through retweeting. There are four connection methods: retweet and retweeted, mention, and mentioned. Characteristics extracted for each connection are mainly related to language, local time and popularity, etc. Additionally, due to Twitter's limits, the model does not use follower/followee information but these aggregate statistics. Network features can be split into 3 networks: retweet, mention, and hashtag co-occurrence network structures. Retweet and mention networks represent users as nodes, and the directed links connecting pairs of users obey the message flow direction. Hashtag co-occurrences network represents hashtags as nodes and undirected links connecting two tags that appear in the same tweet. All networks are weighted based on the frequency of interaction or co-occurrences. Temporal features are extracted by measuring the average degree of Twitter propagation over different periods and the distribution of time intervals between events. Content and language features can be extracted through the Part-of-Speech (POS) technique that can annotate tweets' verbs, nouns, adjectives, modal auxiliaries, pre-determiners, interjections, adverbs, wh-, and pronouns. Statistics based on the number of words and information entropy are also crucial, including min, max, median, mean, and std. deviation, skewness, kurtosis, and entropy. Sentiment features come from the leverage of several sentiment extraction techniques to generate various sentiment features, including arousal, valence, and dominance scores (Cresci et al., 2019), happiness score (Warriner et al., 2013), polarization and strength (Kloumann et al., 2012), and emoticon score (Wilson et al., 2005).

### *4.3 Account Characterization*

Botometer introduces the 'bot score' to quantify how likely one account can be a social robot, which corresponds to the proportion of decision trees in random forests that categorize the account as a bot. It is worth noting that the score is not technically equal to the probability of one account being a social robot but should be interpreted by Complete Automate Probability (CAP) and equal to $P(Bot|Score)$ representing the Bayesian posterior probability of account with this or higher score being a bot. For account with a $0.96/1$ score having a probability of $90\%$ means there is $90\%$ of social bots having a $0.96/1$ score have been detected. The exact equation is shown below.

$$P(Bot|Score) = \frac{P(Bot) \times P(Score|Bot)}{P(Score)}$$

The evidence $P(Score)$ can be derived as:

$$\begin{aligned} P(Score) &= P(Score, Bot) + P(Score, Human) = P(Score|Bot) \times P(Bot) + P(Score|Human) \times P(Human) \\ &= P(Score|Bot) \times P(Bot) + P(Score|Human) \times \big(1 - P(Human)\big) \end{aligned}$$

where can be transformed to obtain $P(Score|Bot)$ and $P(Score|Human)$.

Botometer characterizes friendship ties and information flow through 4 topological networks: friends, followers, mentions, retweets, and exploits K-means, one of the unsupervised learning models, to cluster accounts through features extracted by random forest.

K-means algorithm is an iterative clustering algorithm that divides the data into $K$ groups and selects $K$ samples as the initial cluster centers, calculates the distance, such as Euclidean distance or Manhattan distance, between each sample and each center, and assigns each sample to its nearest center, then a cluster can be formed by centers and the assigned samples. The cluster center will be recalculated iteratively by averaging the samples of each cluster until meeting convergence. The specific algorithm is shown in *Algorithm 2*。

**Algorithm 2:** $k-means(D,k)$

**Data:** $D$ is a dataset of $n$ $d$-dimensional points; $k$ is the number of clusters.

Initialize $k$ centerc $C=[c_1,c_2,\ldots,c_k]$;
$canStop \leftarrow$ **false**;
**while** $canStop =$ ***false*** **do**
  Initialize $k$ empty clusters $G=[g_1,g_2,\ldots,g_k]$;
  **for each** data point $p \in D$ **do**
    $c_x \leftarrow \text{NearestCenter}(p,C)$;
    $g_{c_x}.\text{append}(p)$;
  $LastCluster \leftarrow C$;
  **for each** group $g \in G$ **do**
    $c_i \leftarrow \text{ComputeCenter}(g)$;
  **if** $LastCluster = C$ **then**
    $canStop \leftarrow$ **true**
**return** $G$;

### *4.3.1 Social Connectivity and Information Flow*

As shown in Figure 1, the analysis of social connectivity is mainly studied through the bot score of distribution of account friends and followers and observes that human is usually followed by other human and complicated bots, but bots tend to follow other bots, and they are mostly followed by other bots. Simultaneously, the analysis of information flow based on the bot score of distribution of account mentions and retweets discovers that simple bots normally retweet each other but mention complicated bots, and sophisticated bots might retweet but not mention humans for being unable to communicate with the human. Additionally, human also retweets bot sometimes, for they might publish some interesting content but usually has no interest in mentioning them.

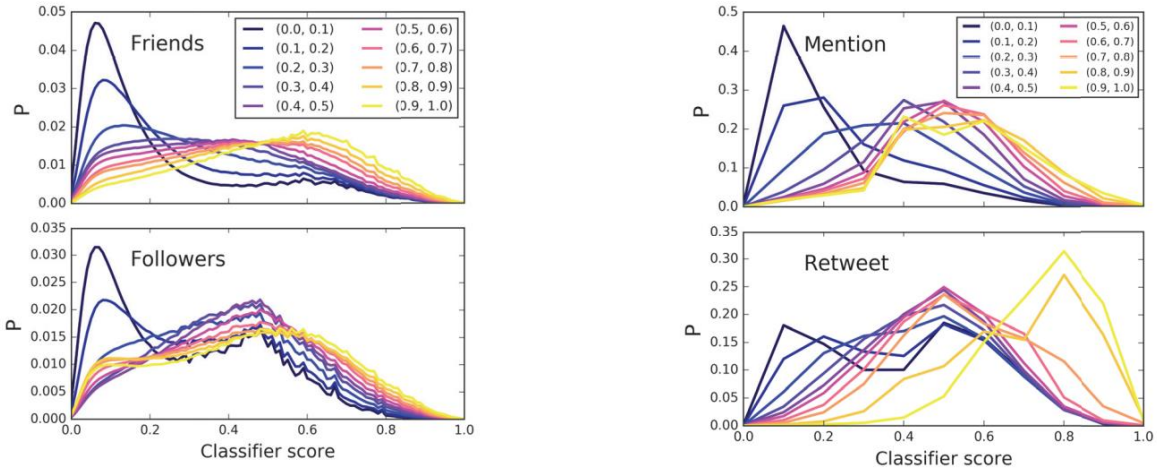


Figure 1: Distribution of bot score for Friends, Followers, Mention and Retweet
Source: Varol et al. (2017).

The conclusion is the account of friendship ties, and information flow explains the various natures of behaviors. On average, human has more interaction and higher reciprocity of friendship with the human-like account, but bots tend to target accounts randomly or intension-oriented.

#### *4.3.2 Account Clustering*

By applying K-means and dimension reduction technique t-SNE (Van der Maaten & Hinton, 2008), The distribution of accounts can be visualized in Figure 2. Three main social bots can be summarized from the demonstration: $C0$ concentrating in the bottom of 2-dimensional embedding consists of self-promote accounts like recruiters, porn-actress, etc. $C1$ contains spam accounts that are very active but have fewer followers. $C2$ includes accounts frequently using automated applications to share activity from other platforms like YouTube and Instagram or post links to news articles. Some of the accounts in $C2$ might belong to actual humans who are no longer active, and their posts are mostly sent by connected apps.

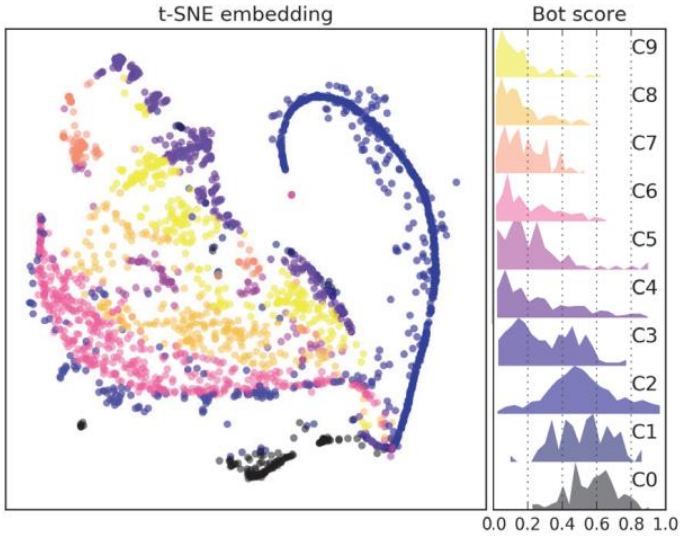


Figure 2: t-SNE embedding of accounts
Source: Varol et al. (2017).

#### *4.4 Ensemble of Specialized Classifier (ESC)*

Botometer deploys an inheritance classifier, which is a random forest dedicated to identifying human accounts combined with multiple other random forests. These forests are used to identify each specific kind of robot account and score through the maximum rule of joint voting to obtain the classification results (Yang et al., 2019), as shown below.

$$winning\ class\ as\ i^* = \arg\max_i\{s'_i\}\ where$$

$$s'_i = \begin{cases} 1 - s_i, if\ i = bot \\ s_i, else \end{cases}$$

## 5. Discussion

Botometer has also exposed vulnerabilities in practice, and the study found that Botometer, in addition to being able to identify fan robots, cannot identify promotional robots, advertising robots, trending topic robots, etc. In this section, we discuss that the main issue of Botometer is data dependency, provide two correspondings, and introduce more possible attempts that have achieved competitive performance.

#### *5.1 Possible Issues*

Various features of social robots are constantly changing, and to ensure efficient and accurate recognition, the data for machine learning needs to be constantly updated. Moreover, the recognition process and effect are subject to feature extraction and classification algorithms, and different data sets may bring different results. At the same time, social media has dynamic characteristics, which puts forward higher requirements for data mining.

Data fed into Botometer has considered adopting tags and tried to quantify which are being amplified by robo-accounts. They used Tweepy, a Python package that helps access the Twitter API, to search for tweets containing these tags. First, let's count the

number of unique accounts in each data set. In all data sets, the number of unique accounts is much smaller than the number of tweets, suggesting that some accounts post the same cash hashtag multiple times. The next step is to query the Botometer API for BOT analysis. Instead of going through every tweet and checking every user they encounter, researchers can keep track of the accounts that have been queried to avoid duplication and increase efficiency. The Botometer API returns a wealth of information about each account. For flexibility, we recommend storing the full results of the Botometer. As mentioned above, the Botometer generates an overall score and a language-independent score. Since the two scores come from different classifiers, they are not comparable and should not be mixed together. To decide which language to use, let's calculate the percentage of accounts that use each language.

The model has misjudgments on social bots and human accounts. The main reason is that tweets are created by platform applications, or retweets account for a high proportion. Misjudgment of official accounts or marketing accounts of organizations: Often, such accounts are co-managed by multiple users or automated programs, and the model tends to classify accounts with multiple languages in their tweets as social bots. At the same time, the dependence of the model on data is also likely to cause the limitations of the model, which is mainly for English tweets and relies on a large number of labeled data, and the recognition degree of unknown types of social robots is low. In this regard, we propose the following optimization suggestions:

- Machine learning systems that focus on online data to make inferences and predictions are challenging. On the one hand, the platform can change functionality and require the model to be retrained. Further difficulties arise when accounts used for training change behavior, become inactive, compromise, or are removed from the platform, invalidating ground truth data. Computing behavior, on the other hand, can change and evolve. As a typical adversarial setup, automated accounts are made more complex to evade detection. The advent of more advanced robotic capabilities creates additional challenges for existing systems that are difficult to generalize to new behaviors. Although when the training and test sets are from the same domain, even with cross-validation, the supervised model misses the new robot class, resulting in a low recall (Yang et al., 2019).
- Use more complex robot scenarios and other robot detection systems than Botometer. While Botometer was able to detect in many cases that the fan bot was only retweeting from a single account, in almost all cases, it was unable to detect multiple social bots. This indicates an urgent need for effective social robot detection models. Explore other factors that affect robot detection performance, such as profile picture, Twitter rate, and biometric information. Create a test set to replace the public data set with robots to reliably evaluate the performance of the robot detection system (Cresci et al., 2017). New social robots are automatically identified by a collection of specialized classifiers trained on different robot classes.

### *5.2 Other Techniques and Models*

In this section, we decide to introduce three successful techniques and models, containing the invention of digital fingerprints as a new feature and neural network support, including deep neural networks and generative adversarial networks.

#### *5.2.1 Digital DNA Technology*

This method (Kudugunta & Ferrara, 2018) analyzes the behavior of accounts, and using digital DNA technology, which is inspired by biology, digital DNA can represent the life cycle of an account's behavior with a series of encoded characters. Designed to socially fingerprint technology, it is capable of identifying human and bot accounts by supervised or unsupervised methods. It is also compared with state-of-the-art detection methods. Finally, our method is able to analyze user behavior using existing DNA analysis techniques and efficiently relies on a limited number of dependent user accounts. The DNA modeling approach focuses primarily on sequences related to action behavior, consisting of sequences of various lengths. Similar to biological DNA with bases (A, C, G, T), digital DNA can be applied to model user and interaction behavior in social networks. Such as tweeting, replying, and following, can be different symbolic codes, and the user's actions represent the primitives of the digital DNA sequence. Encoding someone's behavior in a digital DNA sequence means linking each of the actions one aims to model to a base of the alphabet. Choosing alphabets with different cardinalities represents DNA sequences with different degrees of granularities, for example: $\mathbb{B}^3_{type}$, $\mathbb{B}^3_{content}$ and $\mathbb{B}^6_{content}$

$$\mathbb{B}^3_{type} = \begin{Bmatrix} A \longleftrightarrow tweet \\ C \longleftrightarrow reply \\ T \longleftrightarrow retweet \end{Bmatrix} = \{A, C, T\}$$

$$\mathbb{B}^3_{content} = \begin{Bmatrix} N \longleftrightarrow tweet\ contains\ no\ entities\ (plain\ text) \\ E \longleftrightarrow tweet\ contains\ entities\ of\ one\ type \\ X \longleftrightarrow tweet\ contains\ entities\ of\ mixed\ types \end{Bmatrix} = \{N, E, X\}$$

$$\mathbb{B}^{6}_{content} = \begin{Bmatrix} N \longleftarrow tweet\ contains\ no\ entities\ (plain\ text) \\ U \longleftarrow tweet\ contains\ one\ or\ more\ URLs \\ H \longleftarrow tweet\ contains\ one\ or\ more\ hashtags \\ M \longleftarrow tweet\ contains\ one\ or\ more\ mentions \\ D \longleftarrow tweet\ contains\ one\ or\ more\ medias \\ X \longleftarrow tweet\ contains\ entities\ of\ mixed\ types \end{Bmatrix} = \{N, U, H, M, D, X\}$$

The main superiority of this feature is flexibility and adaptability: While Twitter social bot detection is used on a specific social network, a model based on digital DNA can be a platform-free approach.

#### *5.2.2 Deep Neural Network*

Deep Neural Networks (Lim et al., 2017) have achieved success in social bots detections Kudugunta et al. classify humans and bots from account level and tweet level. For account-level classification, a random forest can reach 98.45%. And use the MOVE method to interpolate in the feature space to generate new samples to better balance the data. The best classification effect achieved by the generated data Adaboost was 99.81%. It turned out that just by performing bot detection at the account level, a simple model could produce a good classification effect. For tweet-level classification, the original content-based detection method did not work very well. Many methods have tried to infer based on content, but this approach has been shown to be ineffective, and in order to overcome the limitations of traditional techniques, long short-term memory models are used. And in order to be able to convert the tweet text into a form suitable for LSTM processing, using an embedding strategy, we used a pre-trained global vector (GloVE) of word representation to characterize the Twitter data.

Moreover, LSTM can be deployed on bots detection as well. The exploration of potential time patterns using CNN-LSTM models with user history tweet data as temporal text data avoids tedious feature engineering (Sayyadiharikandeh et al., 2020). Researchers propose the use of deep learning models to learn representations of social behavior and content to detect social robots. Social behavior has both internal and external causes (internal causes such as people's circadian rhythms, external factors such as weekends and holidays, and people spending more time using social media), and the proposed model learns the social behavior caused by these factors. For the representation of content, extract the time pattern between the contents rather than just extracting the content data. A DBDM model is proposed to capture the underlying characteristics and content information of users' social behavior. DBDM consists of three layers: the input layer, the characterization layer, and the fusion layer. The input layer accepts tweets and timestamps and converts each tweet using word embeddings into a Twitter matrix. The representation layer contains two components; one is the social behavior component, and the other is the time content component. The fusion layer generates the user's representation information by combining information about behavior and content. Add a fully connected layer and a Softmax layer to the top of the fused layer to get the classification label.

#### *5.2.3 Generative Adversarial Networks*

GANs (Cresci et al., 2018) are a powerful class of neural networks that are used for unsupervised learning. It was developed and introduced by Ian J. Goodfellow in 2014. GANs are basically made up of a system of two competing neural network models which compete with each other and are able to analyze, capture and copy the variations within a dataset. It has been noticed most mainstream neural nets can be easily fooled into misclassifying things by adding only a small amount of noise into the original data. Surprisingly, the model, after adding noise, has higher confidence in the wrong prediction than when it was predicted correctly. The reason for such an adversary is that most machine learning models learn from a limited amount of data, which is a huge drawback, as it is prone to overfitting. Also, the mapping between the input and the output is almost linear. Although it may seem that the boundaries of separation between the various classes are linear, in reality, they are composed of linearities, and even a small change in a point in the feature space might lead to the misclassification of data. Generative Adversarial Networks (GANs) can be broken down into generative model, which describes how data is generated in terms of a probabilistic model, adversarial model, which is done in an adversarial setting and using deep neural networks as the artificial intelligence algorithms for training purposes.

## 6. Conclusion

This paper is devoted to doing a documentary analysis of the techniques deployed in bots detections and elaboratively evaluating a public bot detection API, *Botometer*. We first explained the development of social bots and introduced some popular ways of detecting bot accounts, among which we focus on Botometer. Next, we demonstrate that the main attributes of the Botometer are data augmentation, feature engineering, account clustering, and ensemble learning. Then, we discuss the majority of issues that might impair the model performance as data dependency and provide two suggestions, including leveraging online data, deliberating more sophisticated scenarios, and developing more comprehensive models. Finally, we introduce various techniques, including deep learning combined with data re-sampling to relieve expensive annotations and generative adversarial networks combined with digital DNA features to simulate the evaluation of social bots and discover more features to improve model

performance. Social bots have seriously affected the social media ecosystem; thus, the invention of new techniques and the improvement of previous bots detection frameworks are crucial. We will continue to pay attention to these fields, make some efforts and try to advocate more focus and attract more educational investments.

**Funding:** This research received no external funding.
**Conflicts of Interest:** The authors declare no conflict of interest.
**Publisher's Note**: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers.

**References**

[1] Bessi, A., & Ferrara, E. (2016). Social bots distort the 2016 U.S. presidential election online discussion. *First Monday, 21*(11), Article 7090. https://doi.org/10.5210/fm.v21i11.7090
[2] Chen, C. F., & Yuan, Y. Q. (2021). Research on the characteristics and patterns of "computational propaganda" of social robots——Taking China's new crown vaccine issue participation as an example. *Journalism and Writing,* (11), 77–88.
[3] Cresci, S., Petrocchi, M., Spognardi, A., & Tognazzi, S. (2019, 2019/06//). *Better safe than sorry: An adversarial approach to improving social bot detection* [Paper presentation]. Proceedings of the 10th ACM Conference on Web Science, Boston, MA.
[4] Cresci, S., Pietro, R. D., Petrocchi, M., Spognardi, A., & Tesconi, M. (2018). Social fingerprinting: Detection of spambot groups through DNA-inspired behavioral modeling. *IEEE Transactions on Dependable and Secure Computing, 15*(4), 561–576. https://doi.org/10.1109/TDSC.2017.2681672
[5] Cresci, S., Spognardi, A., Petrocchi, M., Tesconi, M., & Pietro, R. D. (2017, 2017). *The paradigm-shift of social spambots: Evidence, theories, and tools for the arms race* [Paper presentation]. Proceedings of the 26th international conference on world wide web companion, Perth.
[6] Fazil, M., & Abulaish, M. (2017, 2017/06//). *Why is a socialbot effective on Twitter? A statistical insight* [Paper presentation]. 2017 9th International Conference on Communication Systems and Networks (COMSNETS), Bengaluru.
[7] Ferrara, E., Varol, O., Davis, C., Menczer, F., & Flammini, A. (2016). The rise of social bots. *Communications of the ACM, 59*(7), 96–104. https://doi.org/10.1145/2818717
[8] Guo, X. A., & Zhao, H. M. (2022). Social robots as 'political ventriloquists': The duality of roles and their transcendence. *Modern Communication (Journal of Communication University of China), 44*(2), 122–131. https://doi.org/10.19997/j.cnki.xdcb.2022.02.016
[9] Howard, P. N., Kollanyi, B., Bradshaw, S., & Neudert, L.-M. (2018). Social media, news, and political information during the US Election: Was polarizing content concentrated in swing states? *arXiv preprint*, arXiv:1802.03573.
[10] Kloumann, I. M., Danforth, C. M., Harris, K. D., Bliss, C. A., & Dodds, P. S. (2012). Positivity of the English language. *PLoS ONE, 7*(1), Article e29484. https://doi.org/10.1371/journal.pone.0029484
[11] Kudugunta, S., & Ferrara, E. (2018). Deep neural networks for bot detection. *Information Sciences, 467*, 312–322. https://doi.org/10.1016/j.ins.2018.08.019
[12] Lee, K., Eoff, B., & Caverlee, J. (2011). *Seven months with the devils: A long-term study of content polluters on Twitter* [Paper presentation]. Proceedings of the international AAAI conference on web and social media, Barcelona.
[13] Lim, E. P., Winslett, M., Sanderson, M., Fu, A., Sun, J., Culpepper, S., Lo, E., Ho, J., Donato, D., Agrawal, R., Zheng, Y., Castillo, C., Sun, A., Tseng, V. S., & Li, C. (2017). *Proceedings of the 2017 ACM conference on information and knowledge management*. ACM.
[14] Liu, R., Chen, B., Yu, L., Liu, Y. S., & Chen, S. Y. (2017). Research on malicious social robot detection technology. *Journal of Communications, 38*(S2), 197–210.
[15] Lu, L. Y., Li, Y. Y., Lu, G. J., Liu, Y., & Wang, C. J. (2021). Social robot-driven computational advocacy: Social robot recognition and analysis of behavioral characteristics. *Journal of Communication University of China (Natural Science Edition), 28*(2), 35–43+53. https://doi.org/10.16196/j.cnki.issn.1673-4793.2021.02.004
[16] Sayyadiharikandeh, M., Varol, O., Yang, K. C., Flammini, A., & Menczer, F. (2020, 2020/10//). *Detection of novel social bots by ensembles of specialized classifiers* [Paper presentation]. Proceedings of the 29th ACM international conference on information & knowledge management, Virtual Event.
[17] Shi, A. B., & Wang, B. (2022). Social robots: Current situation and prospects of news communication in human-machine communication mode. *Young Reporter,* (7), 95–99. https://doi.org/10.15997/j.cnki.qnjz.2022.07.021
[18] Shi, W., & Chen, C. F. (2020). Research on the role and behavior patterns of social robots in news diffusion——Based on the analysis of the New York Times "Amendment" storm report on Twitter. *Journalism and Communication Studies, 27*(5), 5–20+126.
[19] Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research, 9*, 2579–2605.
[20] Varol, O., Ferrara, E., Davis, C., Menczer, F., & Flammini, A. (2017). *Online human-bot interactions: Detection, estimation, and characterization* [Paper presentation]. Proceedings of the International AAAI Conference on Web and Social Media, Atlanta, GA.
[21] Warriner, A. B., Kuperman, V., & Brysbaert, M. (2013). Norms of valence, arousal, and dominance for 13,915 English lemmas. *Behavior Research Methods, 45*(4), 1191–1207. https://doi.org/10.3758/s13428-012-0314-x
[22] Wilson, T., Wiebe, J., & Hoffmann, P. (2005, 2005). *Recognizing contextual polarity in phrase-level sentiment analysis* [Paper presentation]. Proceedings of human language technology conference and conference on empirical methods in natural language processing, Vancouver.
[23] Yang, K. C., Varol, O., Davis, C. A., Ferrara, E., Flammini, A., & Menczer, F. (2019). Arming the public with artificial intelligence to counter social bots. *Human Behavior and Emerging Technologies, 1*(1), 48–61. https://doi.org/10.1002/hbe2.115

[24] Zhang, H. Z., Duan, Z. N., & Han, X. (2019). Heterogeneity or symbiosis: Discussion on the research path of social robots in social media. *Press,* (2), 10–17. https://doi.org/10.15897/j.cnki.cn51-1046/g2.2019.02.002

[25] Zheng, C. Y., & Fan, H. (2020). From social contagion to social diffusion: Research on the social diffusion transmission mechanism of social robots. *Presse Medicale,* (3), 51–62. https://doi.org/10.15897/j.cnki.cn51-1046/g2.20200313.003

[26] Zheng, Q., & Han, N. (2021). Visual analysis of social robot research based on the knowledge graph. *Cyberspace Security, 12*(Z4), 14–24.