Inteligencia Artificial 27(73), 142-158
doi: 10.4114/intartif.vol27iss73pp142-158

INTELIGENCIA ARTIFICIAL

http://journal.iberamia.org/

Unsupervised Machine Learning for Bot Detection on Twitter: Generating and Selecting Features for Accurate Clustering

Raad Ghazi Al-Azawi[1], Safaa O. AL-mamory[2]
[1] Software Department, College of Information Technology, University of Babylon, Babylon, Iraq. raad.alazawi@student.uobabylon.edu.iq
[2] College of Business Informatics, University of Information Technology and Communications, Bagdad, Iraq. salmamory@uoitc.edu.iq

Abstract Twitter is a popular social media platform that is widely used by individuals and businesses. However, it is vulnerable to bot attacks, which can have negative effects on society. Supervised machine learning techniques can detect bots but they require labeled data to differentiate between human and bot users. Twitter generates a significant amount of unlabeled data, which can be expensive to be labeled. This issue can be addressed by exploiting the advantages of unsupervised machine learning techniques, specifically clustering algorithms as such techniques are crucial for managing such kind of data and reducing computational complexity. However, feature selection is necessary for clustering, as some features are more important than others. This study aims to enhance feature reliability, introduce new features, and reduce the proposed model's complexity. This, in turn, can improve bot identification accuracy based on clustering algorithms. The study achieved a Fowlkes-Mallows score of 0.99 in DBSCAN clustering algorithms, including agglomerative hierarchy, k-medoids, DBSCAN, and K-means. This was accomplished by minimizing dataset dimensions and selecting essential features. By employing unsupervised machine learning techniques, Twitter can detect and mitigate bot attacks more efficiently, which can positively impact society.

Keywords: Twitter Bot, Feature selection, Feature extraction, Unsupervised machine learning, Clustering algorithms

1 Introduction

Twitter has become one of the most fascinating platforms for individuals to share their thoughts and viewpoints on a diverse range of subjects. Nevertheless, the rise of automated Twitter accounts, known as bots, has become increasingly prevalent in recent years. These bots can influence public opinion, spread false information, and cause other negative social impacts [1, 2]. Malicious bots are those that disseminate spam content, adware, and malware within the realm of public opinion. According to Twitter's estimates, these bots constitute approximately 8.5% of its user base [3].

ISSN: 1137-3601 (print), 1988-3064 (on-line)

Supervised machine learning models are widely regarded as one of the most effective techniques for bot detection, primarily owing to the substantial volume of data generated by automated Twitter accounts, or bots [4]. These methods differ from analyzing user social behavior as they prioritize statistical attributes or features and the significance of a particular set of differentiating features [5]. Supervised machine learning techniques rely on labeled data for prediction, which is a limitation as real-world Twitter data is mostly unlabeled. Unsupervised models like clustering methods have been developed as a solution, which does not require labeled data to detect bots [6]. Instead, they focus on the similarity between accounts within a single cluster. Therefore, selecting robust and stable features is crucial for the success of cluster algorithms.

The primary of this study is to enhance the performance of clustering algorithms. This is achieved by generating new features and identifying the most critical ones using the Correlation Attribute Evaluates (CAE) technique. The focal challenge of this endeavor is to curtail computational complexity by extracting features exclusively from the metadata of Twitter user accounts. This work adds significant contributions in comparison to previous studies. This includes:

1) Developing more reliable new features that solely use the meta-information of Twitter accounts to assist cluster techniques in detecting bots.
2) Demonstrating that the approach features can be used with four different clustering techniques (agglomerating, k-medoids, DBSCAN, and K-means) to address bot identification challenges caused by missing labels and outliers.
3) By selecting the top-ranking features and reducing dimensions, an accuracy rate of 0.99 was achieved.

2 Related Works

The feature extraction method (FE) aids in diminishing dimensionality and elevating learning accuracy. Twitter metadata furnishes information about diverse events linked to a tweet, encompassing its posting time and location. In the ensuing discourse, we will delve into several topics about the commonly utilized metadata on Twitter.

In a previous research study [7], a set of robust features was created to identify Twitter spammers using a combination of linear regression and PCA. The newly generated feature set improved the detection accuracy and reduced false-positive rates. However, the study also highlighted that when only PCA is used for feature extraction, important information might be overlooked. To address this issue, our study gave each feature a functional score before selecting the features with the highest score. The hybrid LR-PCA technique may not be generalized well across various social media platforms. PCA's dimension reduction may lead to losing important insights. Moreover, noise sensitivity and complexity could affect a model's robustness.

The authors of [8] employed a multi-objective hybrid strategy that employed the Minimum Redundancy – Maximum Relevance (mRMR) algorithm to identify the most effective feature set for detecting fake Twitter accounts. In addition, they used conventional statistical criteria such as entropy and standard deviation to extract other features. Their proposed approach was tested on two Twitter datasets and yielded an accuracy of 98%. However, a potential issue with the mRMR algorithm is the selection of unimportant features, which is more likely to occur during the algorithm's initial iterations.

Another research study [9] utilized an ensemble-learning-based method to analyze a large dataset of tweets related to COVID-19 and validate the credibility of a significant number of tweets. The approach used in this study involved dividing the data into two categories: credible and non-credible. The classification of tweet credibility was based on various factors, including user and tweet-level attributes, which combined 26 custom and generic meta-properties. In this research, some limitations emerged. Although it is effective for COVID-19 misinformation, its relevance could be limited to pandemic-related content due to distinct characteristics. Moreover, the method's adaptability to evolving misinformation scenarios warrants further scrutiny, given the dynamic nature of information dissemination. The study could benefit from addressing potential bias in training data which could be influenced by subjective judgments.

In [10], results showed that spreading false news is ineffective if people are not willing to believe it. The research analyzed Twitter accounts with a high number of bots among their friends and identified them as credulous users. The study found that several meta-features such as the number of tweets, friends, and followers were statistically significant in distinguishing credulous and non-credulous users. Furthermore, the study found that using two statistical tests on credulous users resulted in more bot content being amplified than on non-credulous users when evaluating retweets and replies.

Two studies manually selected features [11, 12], whereas another study [13] proposed that one-class classification can be used to improve Twitter bot detection because it enables the detection of new bot accounts while just requiring samples of legitimate accounts. To define the accounts and distinguish between bots and humans, one-class classifiers have the advantage of not requiring examples of aberrant behavior, which in this case is the behavior of bot accounts. The experiments of this approach revealed that various forms of bots can be reliably detected with more than 0.89 performance. The features included a mix of text, nominal, and numeric data. However, the one-class classifiers were chosen only to deal with numeric data.

3 Research Methodology

3.1 Methodological framework

Figure 1 shows the proposed system. It operates under the assumption that Twitter data in real-world scenarios is not labeled. This system aims to detect bots using unsupervised models, specifically cluster approaches, that do not require labeled data. The key principle behind these models is to identify similarities between accounts in a given cluster. The effectiveness of the predictions generated by these algorithms depends on the data's readiness and the identification of crucial features. Thus, the suggested system is divided into four distinct phases:

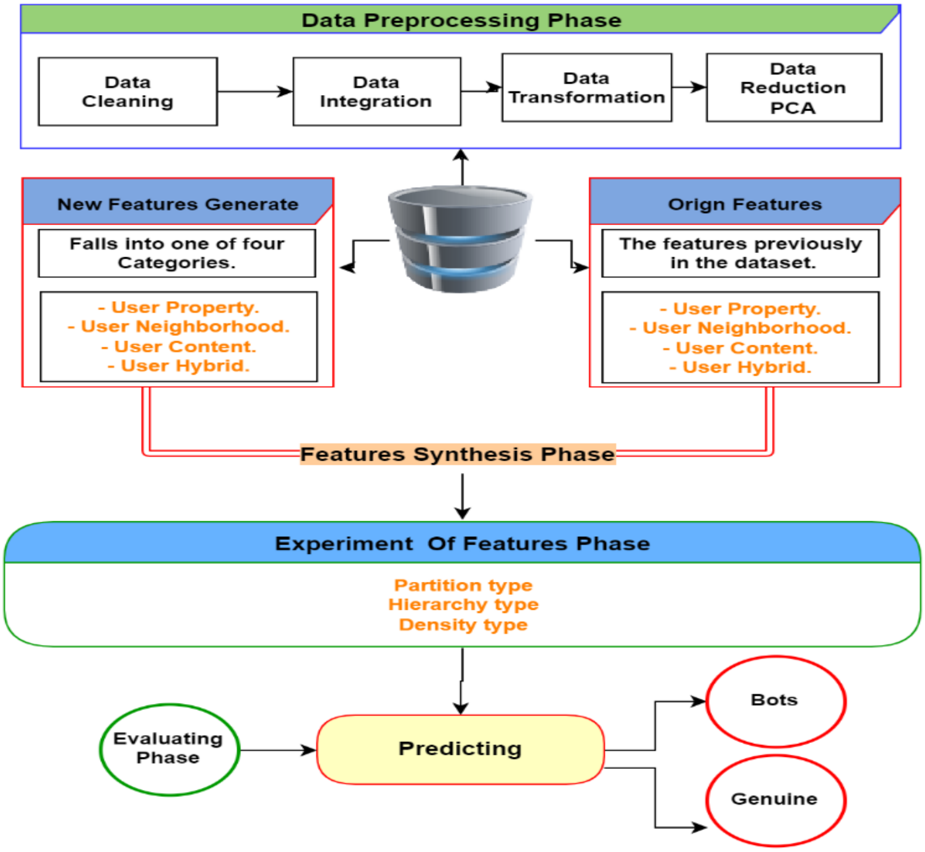


Figure 1. The proposed system.

1) The Preprocessing Phase (Phase One): This involves preparing the data and includes tasks such as data cleaning and formatting.
2) The Feature Enhancement Phase (Phase Two): Here, new features are created, and the best ones are selected to improve clustering algorithms. To expedite the clustering process, Principal Component Analysis (PCA) is employed to reduce data dimensions.
3) The Feature Experimentation and Testing Phase (Phase Three): This phase focuses on testing chosen features using three distinct clustering techniques: partition, hierarchical, and density-based approaches. Four cluster algorithms namely, agglomerative, DBSCAN, K-Means, and k-medoids are employed to effectively handle features using unsupervised machine learning methods.
4) The Evaluation Phase (Final Phase): This phase assesses the performance of the system and interprets the results obtained from the previous phases.

3.2 Preparation of the dataset

3.2.1 Dataset description

Bot detection techniques have been evaluated using various datasets. One of the initial datasets used for bot detection is caverlee-2011[14], which comprises Twitter data collected between December 30, 2009, and August 2, 2010. The dataset consists of 22,223 content polluters, their followers' activity over time, and 2,353,473 tweets. Additionally, the dataset includes 19,276 genuine users, their followers' activity over time, and 3,259,693 tweets. Table 1 provides a summary of the Caverlee dataset for each type of Twitter account, both before and after filtering for English-language tweets. The data includes the number of user-profiles and tweets associated with each account type. The "User Profiles (Before Filtering)" column represents the total number of user profiles for each account type before applying the English language filtering. The "Tweets (Before Filtering)" column shows the total number of tweets for each account type before filtering. The "User Profiles (After Filtering)" column indicates the remaining number of user profiles after filtering for English-language tweets. Lastly, the "Tweets (After Filtering)" column displays the remaining number of tweets after the filtering process.

The decision to use this dataset for testing the proposed system is based on multiple factors. Firstly, the dataset has been widely used in various studies, and most of these studies have utilized supervised machine-learning techniques that are known for their high accuracy and quick implementation. This poses a significant challenge in evaluating the results of the proposed system, which relies on unsupervised machine learning algorithms. The second reason is that the dataset lacks specific features that can effectively differentiate legitimate accounts from malicious bot accounts. Therefore, it is crucial to test the proposed system on this dataset to evaluate its effectiveness in detecting bot accounts using unsupervised machine-learning techniques.

Table 1: Caverlee Dataset Summary - Twitter Accounts Pre/Post English Tweet Filtering

Class	User Profiles (Before Filtering)	Tweets (Before Filtering)	User Profiles (After Filtering)	Tweets (After Filtering)
Polluters	22,223	2,380,059	20,292	2,090,802
Legit Users	19,276	3,263,238	14,180	1,611,205

3.2.2 The Dataset Preprocessing

The first step in preparing data for machine learning algorithms is the preprocessing step, which involves transforming raw data into a comprehensible format. Before applying machine learning models, it is critical to ensure that the data is of good quality. The primary tasks involved in data preprocessing include data cleaning, data integration, data transformation (Standardization, and Normalization), and data reduction using Principal component analysis (PCA). PCA is a statistical technique that transforms high-dimensional data into low-dimensional data by identifying the most important features.

3.3 Generating new features

Feature generation refers to the process of transforming raw input into features suitable for a machine-learning model [15]. Selecting a subset of crucial features is an efficient tactic to manage clusters. It aids in effectively identifying clusters, better comprehending the data, and dimension size reduction for effective storage, collection, and processing [16]. This methodology produced a wide range of features falling under one of four categories.

3.3.1 User Property

This section provides descriptive information about a user and his/her account, including age and Twitter profile information:

1) age_month: The account's age is determined by a monthly value calculated using Formula 1 [17] based on the duration between the time the account was acquired and its creation date.

$$\text{age\_month} = (\text{CreatedYearAt} - \text{CollectedYearAt} * 12) + (\text{number of days}/30.4) \tag{1}$$

where 30.4, is the average duration of a month. The age of a Twitter account is a crucial factor in determining its credibility and identifying malicious bots on the platform. Generally, older accounts are considered more trustworthy than newly-created ones. This metric is utilized in the proposed system to derive additional features like "CV_Following," "FollowingToAgeRate," "Avg_tweets," and "Bfr_afr."

2) age_days: The age of a Twitter account is determined based on the duration between the time it was created and the current date. This age is usually expressed in terms of the number of days that have elapsed since the account's creation. To calculate the daily age of a Twitter account, a specific formula 2 is employed, which takes into account the account's creation date and the current date. This calculation results in a numerical value that represents the daily age of the account, which is then used to determine the account's credibility and authenticity.

$$\text{age\_days} = (\text{CreatedYearAt} - \text{CollectedYearAt}) * 365.24 \tag{2}$$

where 365.24 is the number of days in a year on average, considering the leap years as well.

3) Pro_Info: The profile information feature is used to evaluate the credibility of a Twitter account and it is assigned a numerical value based on the information presented in the account's profile. Typically, four types of information are usually provided by humans when creating a Twitter account, which includes a bio, name, location, and account verification status. In the proposed approach, a value of 1 is assigned to each of these pieces of information if they are present in the account's profile, whereas a value of 0 is assigned if any of these pieces of information are missing.

$$\text{Pro\_Info} = (\text{bio} + \text{name} + \text{location} + \text{verification}) \tag{3}$$

where bio is a biography is a personal description of a profile, It appears that a small number of legitimate accounts leave this feature, whereas a large number of bot accounts leave this feature. Additionally, verification A verified account, according to Twitter, is any account of public interest that has been verified by the company itself, to receive the blue badge, your account must be authentic, notable, and active[18, 19].

3.3.2 User Neighborhood

The most significant indicator of a Twitter account's social influence is typically reflected in its friendship information, which includes the number of followers and following accounts. In this section, five distinct features have been developed to evaluate this aspect of a Twitter account:

1) MaxMinfollowing: The MaxMinfollowing feature is employed to calculate the difference between the maximum and minimum number of following accounts for a Twitter account based on its age. It has been observed that in human accounts, this difference tends to remain relatively stable over time. However, in the case of bot accounts, this difference can be significant and unstable. Thus, the MaxMinfollowing feature can be useful in distinguishing between human and bot accounts.

$$\text{MaxMinfollowing} = [\text{MAX}(\text{UsersFollowings}) - \text{MIN}(\text{UsersFollowings})] \tag{4}$$

2) CV_Following: One of the key features used to assess the credibility of a Twitter account is the CV_Following metric. This feature is calculated using the coefficient of variation (CV) to measure the monthly variability in the number of accounts that user-following. A Twitter account with a high CV value is considered to be unstable and may be classified as a bot. The coefficient of variation is used in this feature because the average number of tweets per user may vary significantly based on the account's age. The equation used to calculate this feature takes into account these differences in tweet averages across accounts as shown in Equation 5.

$$\text{Stander Deviation(Sd)} = \sum_{i=1}^{n} \left(\frac{(\text{Xi} - \text{average})\text{^}2}{\text{MonthsNumbers}} \right) \tag{5}$$

$$\text{Coefficient variance (CV)} = \frac{\text{stander deviation(Sd)}}{\text{average}} \tag{6}$$

$$\text{Average} = \frac{\text{SumMonthFollowings}}{\text{MonthsNumbers}} \tag{7}$$

where Xi is the number-of-following for each month, and the MonthsNumbers is Account age as months.

3) FollowertoFollowingRatio: The objective of deriving the "following_vs_followers_ratio" feature is to indicate the balance between the number of accounts a user follows versus the number of accounts following that user. A well-balanced ratio is typically considered more typical of human accounts, whereas bots may exhibit an imbalanced ratio. Generally, a stable ratio for human accounts is expected to fall within the range of 0.75 to 1.25 [20].

$$\text{FRatio} = \frac{\text{Number of Followers}}{\text{Number of Followings}} \tag{8}$$

4) FollowerToAgeRate: The FollowerToAgeRate feature is computed by calculating the average number of followers gained by a Twitter account per month. This feature is useful in evaluating the credibility of a Twitter account since it indicates the rate at which the account is gaining followers. It has been observed that the follower ratio for human accounts tends to be lower compared to bot accounts. This is because bot accounts may engage in tactics such as mass following and spamming to increase their follower count quickly, whereas human accounts tend to gain followers at a more gradual pace.

$$\text{FollowerToAgeRate} = \frac{\text{NumberofFollowers}}{\text{MonthsNumbers}} \tag{9}$$

5) FollowingToAgeRate: The FollowingToAgeRate feature is used to calculate the average number of accounts-following following a Twitter account per month. This feature is important in evaluating the credibility of a Twitter account since it indicates the rate at which the account is following other accounts. It has been observed that the following ratio tends to be lower for bot accounts compared to human accounts. This is because bot accounts often follow a large number of accounts in a short period to appear more active and gain more followers. In contrast, human accounts tend to follow other accounts at a more moderate pace.

$$\text{FollowingsToAgeRate} = \frac{\text{NumberofFollowings}}{\text{MonthsNumbers}} \tag{10}$$

3.3.3 User Content

Descriptive information about tweets posted information, including the number of tweets, retweets, length of tweets, etc. Four features are created in this section.

1) Avg_tweets: The Avg_tweets feature calculates the average number of tweets posted by a Twitter account per month. This feature is important in determining the credibility of a Twitter account since it indicates the account's activity level. It has been observed that the ratio of tweets from bot accounts to tweets from human accounts is substantially larger. To compute this feature, Equation 11 is used.

$$\text{Avg\_tweets} = \frac{\text{Numbers\_ofTweets}}{\text{MonthsNumbers}} \tag{11}$$

2) bfr_afr: The bfr_afr feature compares the length of the original tweet with the length of the tweet after removing symbols such as hashtags, HTTP links, and other special characters. This feature is important in determining the credibility of a Twitter account since it indicates the linguistic complexity of the tweets posted by the account. It has been observed that bot accounts' alphabetic content is shorter than human accounts. This may be because bots tend to post pre-written or automatically generated content that is less sophisticated than the content generated by humans.

$$\text{bfr} - \text{afr} = \frac{\text{Original tweet} - \text{length of tweet after cleaning}}{\text{MonthsNumbers}} \tag{12}$$

3) MaxOfMonth: MaxOfMonth is a metric that measures the maximum number of tweets posted in a month. This metric can analyze trends and identify peak activity periods on social media platforms. According to observations, the average number of tweets posted by bots was higher than the average number of tweets posted by human users. This suggests that bots may be responsible for a significant portion of the activity on social media platforms.

$$\text{MaxOfMonth} = \text{MAX}(\text{Numbers Monthly Tweets}) \tag{13}$$

4) MaxOfDays: The maximum number of tweets in a day. It has been observed that the average number of human tweets was lower than the number of bots tweets.

$$\text{MaxOfDays} = \text{MAX}(\text{Number of daily tweets}) \tag{14}$$

3.3.4 User Hybrid

This area combines features from a user's property, user's content, and user neighborhood, one feature was created here.

1) By blending the profile information represented by the feature Pro_Info with the percentage of followers of the account, this feature is generated using Equation 15.

$$\text{ProFollow Ratio} = \text{Ln}\left(\left(\frac{\text{FollowersNumbers}}{\text{FollowingsNumbers}}\right) + \text{Pro}\right) * 100 \tag{15}$$

This feature discusses a method for more accurately distinguishing Twitter bots and legitimate accounts by incorporating additional information from the account's profile. This information includes the account's bio, name, location, and whether or not it is verified. By combining this profile information with the percentage of followers, the method aims to provide a more accurate indicator of the account's credibility. The use of profile information is a common approach to distinguishing bots and genuine social media accounts. Bots often have generic or vague profile information, while legitimate accounts tend to have more personalized and detailed information. Incorporating profile information into social media analysis can help identify accounts likely to be bots. Additionally, considering the percentage of followers as an indicator of accreditation can improve accuracy. High percentages of fake or low-quality followers suggest that the account may be a bot. Conversely, genuine accounts often have a higher proportion of engaged followers. Combining profile information and follower percentages can provide a more comprehensive understanding of an account's credibility.

3.4 Feature Selection Technique

In pursuance of obtaining the essential features that support bot detection, the features previously in the database were gathered and named the "original features" alongside the proposed features. Table 2 shows the details of these features based on the category that has been represented. Under the authority of the Correlation Attribute Eval (CAE) technique [21], these features are ranked and only positive rank features are chosen for an account. Table 3 shows the ranking of these features.

Table 2: The details of features with the group you represent

Id	Taxonomy	Feature name	description
1	User Property	LengthOfScreenName	Original features
2	User Property	LengthOfDescriptionInUserPro	Original features
3	User Property	CreatedAt	Original features
4	User Property	CollectedAt	Original features
5	User Neighborhood	NumerOfFollowings	Original features
6	User Neighborhood	NumberOfFollowers	Original features
7	User Content	NumberOfTweets	Original features
8	User Neighborhood	CV_Following	Proposed Features
9	User Neighborhood	FollowertoFollowingRatio	Proposed Features
10	User Hybrid	ProFollow Ratio	Proposed Features
11	User Hybrid	FollowerToAgeRate	Proposed Features
12	User Hybrid	FollowingToAgeRate	Proposed Features
13	User Content	Avg_tweets	Proposed Features
14	User Content	bfr-afr	Proposed Features
15	User Content	MaxOfMonth	Proposed Features
16	User Content	MaxOfDays	Proposed Features

17	User Property	age_month	Proposed Features
18	User Property	age_days	Proposed Features
19	User Property	Max-Min	Proposed Features
20	User Property	Pro	Proposed Features

Table 3: Ranking of attributes concerning the Correlation Attribute Eval (CA) method

Index	Ranked attributes	Attributes name
11	0.39513	FollowingToAgeRate
18	0.36396	max-min
17	0.35699	CV_Following
5	0.20985	NumerOfFollowings
1	0.16617	LngthOfScreenName
8	0.13545	FollowingtoFollowerRatio
2	0.12405	LengthOfDescriptionInUserProle
12	0.07547	FollowerToAgeRate
6	0.03236	NumberOfFollowers
14	0.00335	AvarageDiffrenceTweetRatio
20	0.00323	bfr-afr
7	-0.01787	FollowertoFollowingRatio
19	-0.05244	CV_Months
15	-0.09551	MaxOfDays
13	-0.1005	No_oftweets/accountage(month)
16	-0.10284	MaxOfMonth
10	-0.1235	NumberOfTweets
3	-0.18584	age_month
4	-0.1913	age_days
9	-0.58003	ProFollowRatio

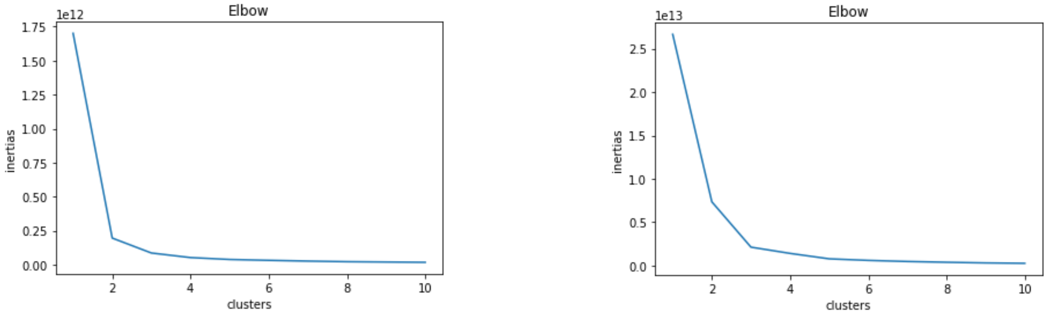
4 Results and Discussion

This section describes the methodology followed by the researchers to test their proposed clustering technique. It will outline the experimental setup, including the data collection process, feature extraction, and clustering algorithms used. The section will also highlight the key findings and results of the experiments.

4.1 Structure of the data

To gain a deeper understanding of the data structure within the Caverlee dataset, the elbow method was employed as shown in figure 2, which is a commonly used technique in unsupervised learning. This method utilizes the K-Means clustering algorithm to determine the optimal number of clusters for the data. By applying this technique, it was aimed to assess the impact of the selected features on the data structure. The analysis involved implementing the K-Means algorithm with different cluster numbers, specifically 11 in this case. Then, the performance of each clustering solution was evaluated by examining the error rate and the Silhouette Coefficient score. The error rate represents the extent to which the data points deviate from their assigned clusters, while the Silhouette Coefficient measures the degree of the separation between the clusters. After conducting the analysis, it was observed that selecting either 2 or 3 clusters resulted in the lowest error rate. Furthermore, these solutions exhibited a high Silhouette Coefficient score of 0.98, indicating well-separated clusters. This suggests that the data points within these clusters were distinct from each other, implying the presence of clear patterns or characteristics.

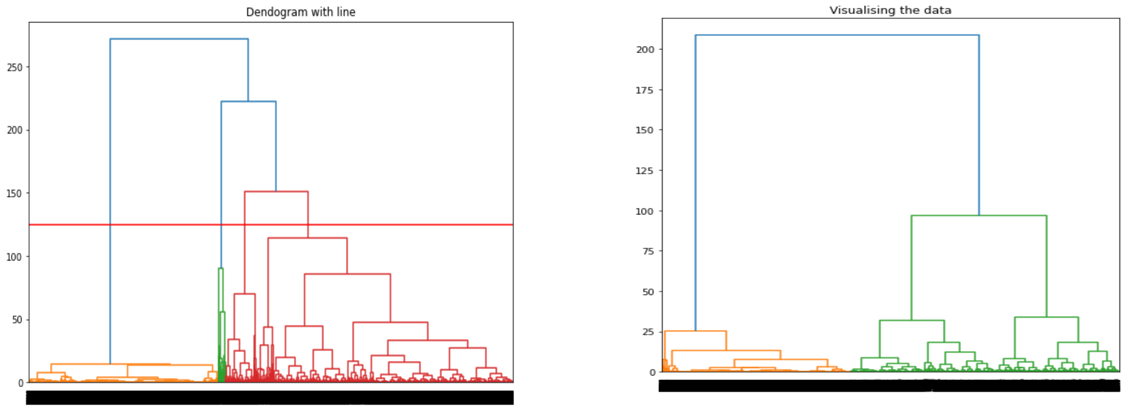
Based on these findings, we can infer that clustering techniques, such as K-Means, are valuable in identifying and distinguishing bots from legitimate users. By leveraging these methods, it becomes possible to discern meaningful groupings within the data, which can aid in the detection of bots by differentiating them from genuine users. The proposed system also analyzed the data using a dendrogram. A key element in interpreting a dendrogram is paying attention to the level at which two objects are linked. The height in the dendrogram reflects the sequence in which clusters were merged. Figure 3 provides a dendrogram where the heights signify the distances between clusters. The dendrogram demonstrates that using origin features (figure 3.a.) results in a different number of clusters compared to using approach features (figure 3.b.).



a. No. of appropriate clustering with 17,388 rows of datset

b. No. of appropriate clustering with 34,776 rows of datset

Figure 2. Elbow Method For Optimal Cluster Numbers Selection.



a- Three numbers of appropriate clustering with old features

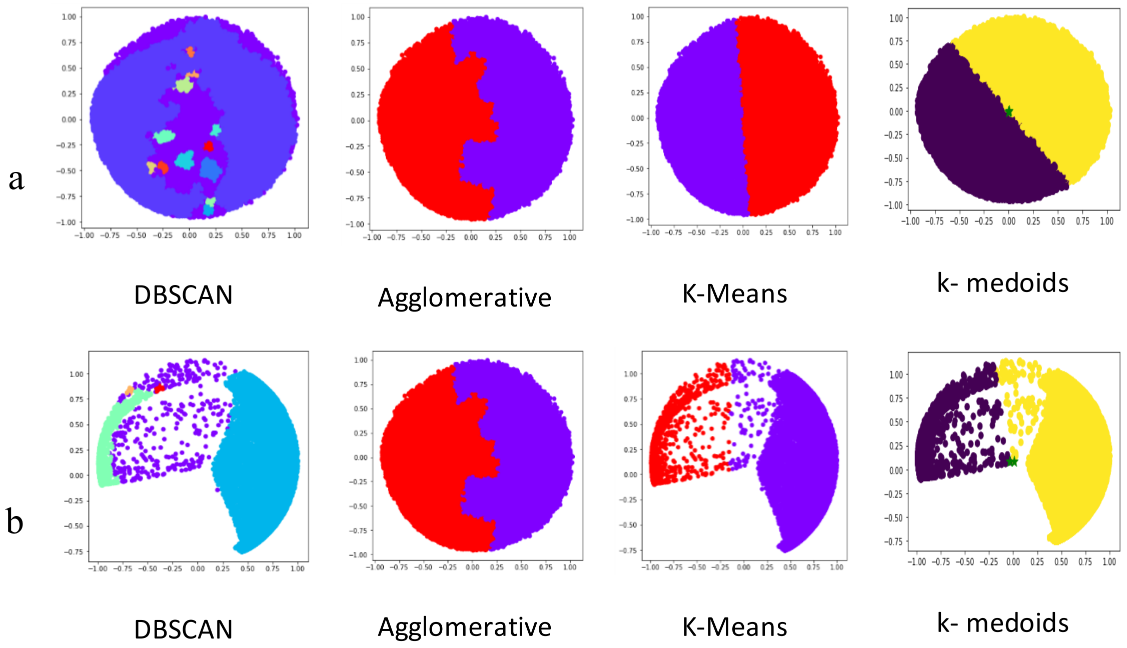
b- Two numbers of appropriate clustering with selected features

Figure 3. Hierarchical dendrogram for clusters and sub-clusters with our points.

4.2 Evaluating the Quality of Clustering

The investigation focuses on the evaluation of the effectiveness of cluster algorithms in detecting bots with the help of the proposed features. The proposed system implemented three types of cluster algorithms: partition, hierarchy, and density. These algorithms are represented by four different techniques, namely k-mean, k-medoid, agglomerative, and DBSCAN. The system used the original features initially, and then the proposed features to evaluate the model using the four techniques mentioned above. The results obtained from each algorithm are then compared to determine which one obtained the highest accuracy. The outcomes of this method are presented in figure 4 which shows the final results of four cluster algorithms. Figure 4.a. displays the outcomes obtained using the original features, while figure 4.b. shows the results obtained using the proposed features. The results indicate that the proposed feature selection improved the performance of the cluster algorithms

The results presented in table 4 demonstrate the efficacy of the proposed features in enhancing the accuracy of the four cluster algorithms employed in this study. The main result indicates that the proposed features have surpassed the original features in terms of accuracy, with a minimum improvement of 0.306 and a maximum improvement of 0.471 across the algorithms. The improvement range was calculated by deducting the original accuracy score from the accuracy score achieved through the proposed features for each algorithm. For instance, the Agglomerative algorithm achieved an accuracy improvement of 0.471, as calculated by subtracting 0.525 from 0.996. Similarly, the k-medoids algorithm achieved an improvement of 0.306, DBSCAN an improvement of 0.378, and K-Means an improvement of 0.469. Therefore, the range of improvement obtained by utilizing the proposed features was between 0.306 and 0.471.



DBSCAN Agglomerative K-Means k- medoids

DBSCAN Agglomerative K-Means k- medoids

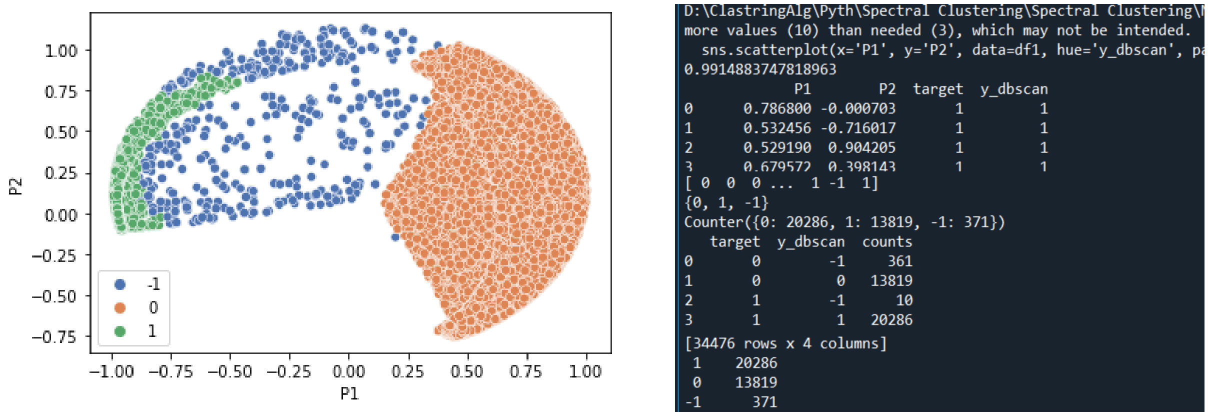
Figure 4. Comparative Results of Cluster Algorithms with Original and Proposed Features.

Table 4: A Study on the Accuracy of the Selected Cluster Algorithms Using Fowlkes-Mallows Scores

clusteringAlg.	parameters	Accuracy with origin features	Accuracy with proposed features
Agglomerative	n_clusters = 2 , affinity='euclidean', linkage='ward'	0.525	0.996
k- medoids	starting_medoids=randome,k=2	0.591	0.897
DBSCAN	eps = 0.0395, min_samples = 50	0.613	0.991
K-Means	n_clusters=2, init='k-means++', random_state=42	0.525	0.994

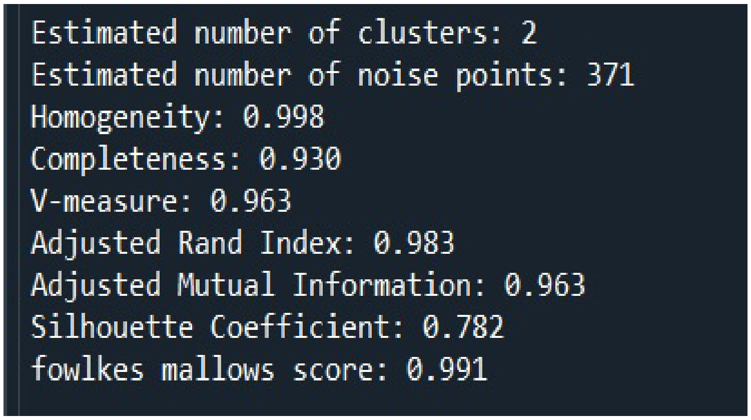
These results demonstrate that the proposed features are more effective in detecting bots than the original features. The study's findings can be useful for improving bot detection techniques and developing more accurate models that can detect bots more efficiently. However, further research must validate and generalize these results on other datasets and real-world scenarios.

To illustrate how the new features help distinguish bots from humans, the DBSCAN algorithm efficiently clustered extensive datasets into two distinct clusters (Cluster 0: 13,819 points, Cluster 1: 20,286 points), with 361 outliers (Cluster -1). This effectiveness was demonstrated in figure 5. Figure 6 shows the performance evaluation using seven metrics (Homogeneity, Completeness, V-measure, Adjusted Rand Index, Adjusted Mutual Information, Silhouette Coefficient, and Fowlkes-Mallows score) showed consistently high scores: e.g., Homogeneity: 0.998, Completeness: 0.930, V-measure: 0.963, etc. These results underline DBSCAN's accuracy and reliability in clustering tasks in comparison with supervised algorithms, despite its unsupervised nature.



```
D:\ClastringAlg\Pyth\Spectral Clustering\Spectral Clustering\
more values (10) than needed (3), which may not be intended.
  sns.scatterplot(x='P1', y='P2', data=df1, hue='y_dbscan', pa
0.9914883747818963
             P1        P2  target  y_dbscan
0      0.786800 -0.000703       1         1
1      0.532456 -0.716017       1         1
2      0.529190  0.904205       1         1
3      0.679572  0.398143       1         1
[ 0  0  0 ...  1 -1  1]
{0, 1, -1}
Counter({0: 20286, 1: 13819, -1: 371})
   target  y_dbscan  counts
0       0        -1     361
1       0         0   13819
2       1        -1      10
3       1         1   20286
[34476 rows x 4 columns]
 1    20286
 0    13819
-1      371
```

Figure 5. Matching Process between Actual and Predicted Labels by DBSCAN Algorithm



```
Estimated number of clusters: 2
Estimated number of noise points: 371
Homogeneity: 0.998
Completeness: 0.930
V-measure: 0.963
Adjusted Rand Index: 0.983
Adjusted Mutual Information: 0.963
Silhouette Coefficient: 0.782
fowlkes mallows score: 0.991
```

Figure 6. Performance evaluation using seven metrics

4.3 Comparing the results with a supervised technique

The results presented in table 5 show that the proposed methodology significantly enhances the accuracy of the four cluster algorithms in detecting bots. Notably, the addition of the proposed features to the original training data consistently improves the performance of the cluster algorithms. Moreover, the proposed approach performs almost

10% better in unsupervised learning algorithms when it is compared to the supervised techniques used in previous studies, as shown in table 5. This finding highlights the significant contribution of this study towards improving bot detection using unsupervised learning algorithms.

Table 5: Results of the previous study on the same dataset, which used supervised techniques

Ref	Technique	Type	Accuracy
[22]	RandomForest algorithm	supervised	95
[23]	Random Forest Accuracy	supervised	98.42
[24]	deep-learning model	supervised	95
[22]	honeypot model	With supervised	85
[25]	SVM, Naïve Bayes and Improved Support Vector Machine (ISVM)	supervised	90
[26]	Neural Network (RNN)	supervised	92
[27]	k-Nearest Neighbors	supervised	93
[28]	Random Forest	supervised	57.139
	The proposed method	unsupervised	99.86

The superiority of the proposed technique in bot detection using clustering algorithms, particularly stream-based clustering algorithms, is demonstrated in table 6 and compared to several similar studies.

Table 6: Performance Comparison of Bot Detection Methods Based on Stream Clustering

Ref	Performance	Method	Details
[29]	F1 = 0.88	DBScan, K-means++	Uses a clustering approach to find groups based on features of either tweet account or account usage.
[30]	F= 64.1, R =92.9	Incremental Naïve Bayes-DenStream	The INB-DenStream method categorizes tweets into spam and non-spam clusters, utilizing Naïve Bayes to capture the mean and boundary of microclusters.
	F= 63.7, R=92.5	DenStream	
	F=59.6, R=53.4	StreamKM++	
	F=31.5, R=23.7	CluStream	
[31]	F 1 = 0.87	HDBSCAN	Employs the HDBSCAN to detect bots by analyzing their previous patterns of retweeting.
[32]	Purity = 0.9	Uniform Manifold Approximation and Projection, followed by HDBSCAN	A clustering algorithm is utilized to comprehend the features of various kinds of accounts belonging to Twitter's state-sponsored trolls.
[33]	Modularity=0.182	Evolutionary DBSCA	A system designed for additional health monitoring of COVID-19 utilizes the DBSCAN and Louvain method to identify communities in temporal networks.
Proposed method	Fowlkes Mallows Score=99.86	DBSCAN	

5 Conclusion

This study presented a notable advancement in comparison to previous research by adding several different contributions. These include the development of robust new features based solely on the meta-information of Twitter accounts and enhancing the accuracy of cluster techniques in detecting bots. The research showcases the versatility of these features by applying them to four distinct clustering techniques namely, agglomerating, k-medoids, DBSCAN, and K-means. This also helped address challenges associated with bot identification in scenarios involving missing labels and outliers. Furthermore, the study achieved a remarkable accuracy rate of 0.99 through the selection of top-ranking features and dimension reduction. By focusing on Twitter bot detection, the study not only employed clustering techniques but also innovated by extracting new features that augment the precision of clustering algorithms while reducing the dataset's dimensions. It critically examined the value of Principal Component Analysis (PCA) due to potential information loss, optimized its implementation by utilizing a minimal set of five features, and prioritized different feature groups. The application of these features across multiple clustering algorithms yielded an impressive average accuracy rate of 0.9695. However, some limitations have arisen. First, the lack of in-depth exploration of selected feature constraints might influence the research outcomes. Second, the study used a single dataset and this, in turn, has the potential of limiting the findings' applicability. Thus, this research invites conducting further studies to address these shortcomings.

Declarations section

Ethical Approval:

I conducted this study in accordance with ethical principles and regulations, and I ensured the confidentiality and privacy of the participants. We confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed. We further confirm that the order of authors listed in the manuscript has been approved by all of us.

Availability of supporting data: "Not Applicable"

Competing interests:

We wish to draw the attention of the Editor to the following facts which may be considered as potential conflicts of interest and to significant financial contributions to this work. We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

We confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed. We further confirm that the order of authors listed in the manuscript has been approved by all of us.

We confirm that we have given due consideration to the protection of intellectual property associated with this work and that there are no impediments to publication, including the timing of publication, with respect to intellectual property. In so doing we confirm that we have followed the regulations of our institutions concerning intellectual property.

We understand that the Corresponding Author is the sole contact for the Editorial process (including Editorial Manager and direct communications with the office). He is responsible for communicating with the other authors about progress, submissions of revisions and final approval of proofs. We confirm that we have provided a current, correct email address which is accessible by the Corresponding Author.

1- **Raad Ghazi Hameed Al-Azawi Date 10/04/2023**
2- **Safaa O. AL-mamory Date 10/04/2023**

Funding: "Not Applicable"

I would like to clarify that this publication and there has been no significant financial support for this work that could have influenced its outcome.

Authors' contributions:

The Authors' Contributions. This is achieved through:

1) Developing more reliable new features that solely use the meta-information of Twitter accounts to assist cluster techniques in accurately detecting bots.

2) Demonstrating that the approach features can be used with four different clustering techniques (agglomerating, k-medoids, DBSCAN, and K-means) to address bot identification challenges caused by missing labels and outliers.

3) By selecting the top-ranking features and reducing dimensions, an accuracy rate of 0.99 was achieved.

Acknowledgments:

I would like to thank "Stefano Cresci", and "Linwan Wu" for their assistance with data collection and analysis. Finally, I would like to acknowledge the contribution of my adviser, Safaa O. AL-mamory, for their guidance and support throughout this project.

References

[1] M. Jiang, P. Cui, and C. Faloutsos, “Suspicious Behavior Detection: Current Trends and Future Directions,” *IEEE Intell. Syst.*, vol. 31, no. 1, pp. 31–39, 2016, doi: 10.1109/MIS.2016.5.

[2] Z. Chen, R. S. Tanash, R. Stoll, and D. Subramanian, “Hunting malicious bots on twitter: An unsupervised approach,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10540 LNCS, no. November, pp. 501–510, 2017, doi: 10.1007/978-3-319-67256-4_40.

[3] V. S. Subrahmanian *et al.*, “The DARPA Twitter Bot Challenge,” *Computer (Long. Beach. Calif).*, vol. 49, no. 6, pp. 38–46, 2016, doi: 10.1109/MC.2016.183.

[4] E. Alothali, N. Zaki, E. A. Mohamed, and H. Alashwal, “Detecting Social Bots on Twitter: A Literature Review,” *Proc. 2018 13th Int. Conf. Innov. Inf. Technol. IIT 2018*, pp. 175–180, 2019, doi: 10.1109/INNOVATIONS.2018.8605995.

[5] R. R. Rout, G. Lingam, and D. V. L. N. Somayajulu, “Detection of Malicious Social Bots Using Learning Automata with URL Features in Twitter Network,” *IEEE Trans. Comput. Soc. Syst.*, vol. 7, no. 4, pp. 1004–1018, 2020, doi: 10.1109/TCSS.2020.2992223.

[6] R. G. Al-Azawi and S. O. Al-Mamory, “Feature extractions and selection of bot detection on Twitter: A systematic literature review,” *Intel. Artif.*, vol. 25, no. 69, pp. 57–86, 2022, doi: 10.4114/intartif.vol25iss69pp57-86.

[7] N. S. Murugan and G. U. Devi, “Feature extraction using LR-PCA hybridization on twitter data and classification accuracy using machine learning algorithms,” *Cluster Comput.*, vol. 22, pp. 13965–13974, 2019, doi: 10.1007/s10586-018-2158-3.

[8] R. R. Rostami and S. Karbasi, “Detecting fake accounts on twitter social network using multi-objective hybrid feature selection approach,” *Webology*, vol. 17, no. 1, 2020, doi: 10.14704/WEB/V17I1/A204.

[9] M. S. Al-Rakhami and A. M. Al-Amri, “Lies Kill, Facts Save: Detecting COVID-19 Misinformation in Twitter,” *IEEE Access*, vol. 8, pp. 155961–155970, 2020, doi: 10.1109/ACCESS.2020.3019600.

[10] A. Balestrucci, R. De Nicola, M. Petrocchi, and C. Trubiani, “A Behavioural Analysis of Credulous Twitter Users,” 2021.

[11] N. Patel and M. Panchal, “Survey of Feature-based Bot Detection Methodologies,” vol. 2020, pp. 1–4, 2020.

[12] P. G. Pratama and N. A. Rakhmawati, “Social bot detection on 2019 Indonesia president candidate’s supporter’s tweets,” *Procedia Comput. Sci.*, vol. 161, pp. 813–820, 2019, doi: 10.1016/j.procs.2019.11.187.

[13] J. Rodríguez-Ruiz, J. I. Mata-Sánchez, R. Monroy, O. Loyola-González, and A. López-Cuevas, “A one-class classification approach for bot detection on Twitter,” *Comput. Secur.*, vol. 91, 2020, doi:

10.1016/j.cose.2020.101715.

[14] K. Lee, B. D. Eoff, and J. Caverlee, “Seven Months with the Devils: A Long-Term Study of Content Polluters on Twitter,” *Icwsm 2011*, no. January 2011, pp. 185–192, 2006.

[15] James Elderfield, “Feature generation from tweets. Preparing tweets for machine learning,” *Medium*, 2018. https://medium.com/@jad.elderfield/feature-generation-from-tweets-9af0565ad6e6 (accessed Jan. 18, 2023).

[16] M. Dash and H. Liu, “Feature Selection for Clustering,” *Curr. Issues New Appl. 4th Pacific-Asia Conf. PAKDD 2000 Kyoto*, vol. volume 180, no. PAKDD 2000, p. pp 110–121, 2003, [Online]. Available: https://link.springer.com/chapter/10.1007/3-540-45571-X_13

[17] K. Rafay, “Age in days,” *Omni is a Polish Company*, 2022. https://www.omnicalculator.com/everyday-life/age-in-days

[18] “What is a verified Twitter account and how to get one - The Verge.” https://www.theverge.com/23199155/verified-twitter-account-how-to (accessed Nov. 15, 2022).

[19] “What does Twitter verification really mean? And what may happen to it? - MarketWatch.” https://www.marketwatch.com/story/what-does-twitter-verification-really-mean-11667495004 (accessed Jan. 18, 2023).

[20] N. SCHAFFER, “Twitter Following vs Followers: What is the Ideal Ratio?,” *NEAL SCHAFFER*, 2023. https://nealschaffer.com/twitter-followers-following-quality-or-quantity/ (accessed Nov. 16, 2023).

[21] D. Gnanambal, D. Thangaraj, Meenatchi V T, and D. Gayathri, “Classification Algorithms with Attribute Selection: an evaluation study using WEKA,” *Int. J. Adv. Netw. Appl.*, vol. 09, no. 06, pp. 3640–3644, 2018.

[22] O. Varol, E. Ferrara, C. A. Davis, F. Menczer, and A. Flammini, “Online Human-Bot Interactions : Detection , Estimation , and Characterization,” no. Icwsm, pp. 280–289, 2017.

[23] K. Lee, B. D. Eoff, and J. Caverlee, “Seven Months with the Devils: An in-depth characterisation of Bots and Humans on Twitter,” *Fifth Int. AAAI Conf. Weblogs Soc. Media*, pp. 185–192, 2011, [Online]. Available: http://arxiv.org/abs/1704.01508

[24] Jan Novotny, “Twitter bot Detection & Categorization,” 2019.

[25] A. K. Ojo, “Improved Model for Detecting Fake Profiles in Online Social Network: A Case Study of Twitter,” *J. Adv. Math. Comput. Sci.*, vol. 33, no. 4, pp. 1–17, 2019, doi: 10.9734/jamcs/2019/v33i430187.

[26] F. Wei and U. T. Nguyen, “Twitter bot detection using bidirectional long short-term memory neural networks and word embeddings,” *Proc. - 1st IEEE Int. Conf. Trust. Priv. Secur. Intell. Syst. Appl. TPS-ISA 2019*, pp. 101–109, 2019, doi: 10.1109/TPS-ISA48467.2019.00021.

[27] M. Li, H. Wang, L. Yang, Y. Liang, Z. Shang, and H. Wan, “Fast hybrid dimensionality reduction method for classification based on feature selection and grouped feature extraction,” *Expert Syst. Appl.*, vol. 150, no. February, p. 113277, 2020, doi: 10.1016/j.eswa.2020.113277.

[28] S. Cresci, F. Lillo, D. Regoli, S. Tardelli, and M. Tesconi, “$FAKE: Evidence of spam and bot activity in stock microblogs on twitter,” *12th Int. AAAI Conf. Web Soc. Media, ICWSM 2018*, no. Icwsm, pp. 580–583, 2018.

[29] Z. Miller, B. Dickinson, W. Deitrick, W. Hu, and A. H. Wang, “Twitter spammer detection using data stream clustering,” *Inf. Sci. (Ny).*, vol. 260, pp. 64–73, 2014, doi: 10.1016/j.ins.2013.11.016.

[30] H. Tajalizadeh and R. Boostani, “A Novel Stream Clustering Framework for Spam Detection in Twitter,” *IEEE Trans. Comput. Soc. Syst.*, vol. 6, no. 3, pp. 525–534, 2019, doi: 10.1109/TCSS.2019.2910818.

[31] M. Mazza, S. Cresci, M. Avvenuti, W. Quattrociocchi, and M. Tesconi, “RTbust: Exploiting temporal patterns for botnet detection on twitter,” *WebSci 2019 - Proc. 11th ACM Conf. Web Sci.*, pp. 183–192, 2019, doi: 10.1145/3292522.3326015.

[32] M. Mazza, M. Avvenuti, S. Cresci, and M. Tesconi, “Investigating the difference between trolls, social bots, and humans on Twitter,” *Comput. Commun.*, vol. 196, no. December 2021, pp. 23–36, 2022, doi: 10.1016/j.comcom.2022.09.022.

[33] H. Elgazzar, K. Spurlock, and T. Bogart, “Evolutionary clustering and community detection algorithms for social media health surveillance,” *Mach. Learn. with Appl.*, vol. 6, no. March, p. 100084, 2021, doi: 10.1016/j.mlwa.2021.100084.