Robust Algorithm for Identifying Bot Authors on Social Media

Bitao Jin

https://github.com/Bitao2/CSCI2952Q-Final-Project.git

Abstract This study explores machine learning methods for identifying bot accounts on social media platforms, using Twitter as an example, and examines their robustness. With the increase in user-generated content, the existence of fake accounts negatively impacts public opinion and the authenticity of information. Therefore, accurately identifying these accounts is crucial.

The research first compares the performance of different machine learning models on a Twitter dataset and analyzes the account identification effectiveness based on user dimensions and tweet dimensions. The results indicate that considering user behavior information significantly improves the identification accuracy of bot accounts. Additionally, the study investigates common bot disguise techniques and their impact on the robustness of detection systems, finding that user-based detection methods exhibit relatively stable performance under various disguise strategies.

Research Background

In recent years, the rise of social media has led to an increase in user-generated content. While this has enriched online interactions, it has also given rise to the issue of fake accounts. Numerous social media platforms are home to a significant number of automated or semi-automated bot users. These bots can have a range of negative effects, including manipulating public opinion and spreading false or harmful information. Accurately identifying these bots is crucial for maintaining the safety and stability of social media environments. Aljabri et al. 2023

In real-world scenarios, the creators of these bots often employ various methods to evade detection by researchers, creating a common adversarial landscape. Bot developers continually refine their strategies to make their bots resemble ordinary users, while researchers work tirelessly to enhance detection techniques to distinguish these disguised bots from genuine users. Consequently, this back-and-forth between detection and evasion leads to an ongoing evolution within the field, raising the bar for robustness in detection models.

This project will focus on analyzing Twitter data to study the performance of relevant detection models within this dataset.

Data Description

The current project utilizes the dataset provided by https://github.com/rrsr28/Twitter-Bot-Detection/tree/main/Datasets). The dataset was gathered based on the research conducted by Alarfaj et al. 2023. It comprises a total of 155,758 tweets from 92 Twitter accounts collected over a period from 2011 to 2018. Out of these, there are 47 bot accounts and 45 human accounts, with an average of 1,673 tweets per account.

A comparison of the number and posting frequency of bot and human accounts is illustrated in Figure 1. It is evident that the tweeting frequency is consistent between the two groups, indicating that, solely based on the volume of tweets, these bot accounts do not exhibit any significant irregularities.

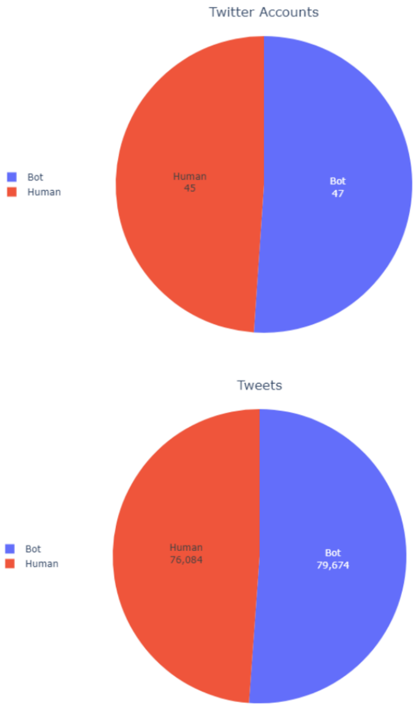

FIGURE 1. *The Accounts Distribution*

The distribution of tweet counts from two groups of users over different time periods is shown in Figure 2. It is evident that prior to 2018, both groups had relatively low posting activity. This is likely due to the data collection period being closer to 2018, making it more challenging to gather complete posting data from earlier years, resulting in certain data gaps.

From the data collected in 2018, it can be observed that the number of tweets sent by Bot users fluctuates less over time compared to human users. This aligns with the nature of Bot behavior, which tends to post automatically.

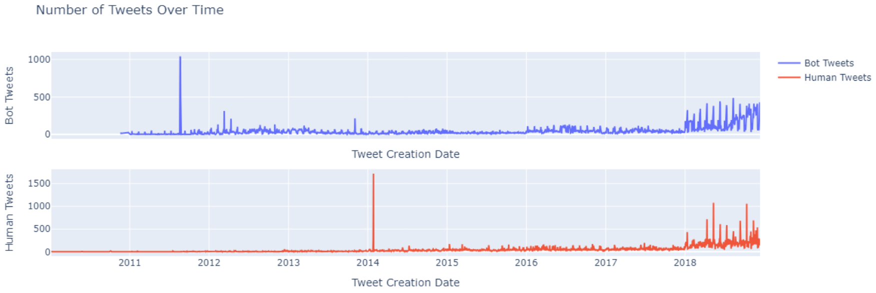

FIGURE 2. *The Tweets Over Time*

The semantic sentiment analysis of tweets from two groups of users is illustrated in Figure 3. From the perspective of tweet sentiment, the most significant difference between the two user groups is that Bot users tend to post a much higher proportion of neutral tweets compared to human users. This indicates a substantial difference in the content shared by these two groups. Therefore, a suitable strategy for identifying Bot users on Twitter is to focus on analyzing the content of their tweets.

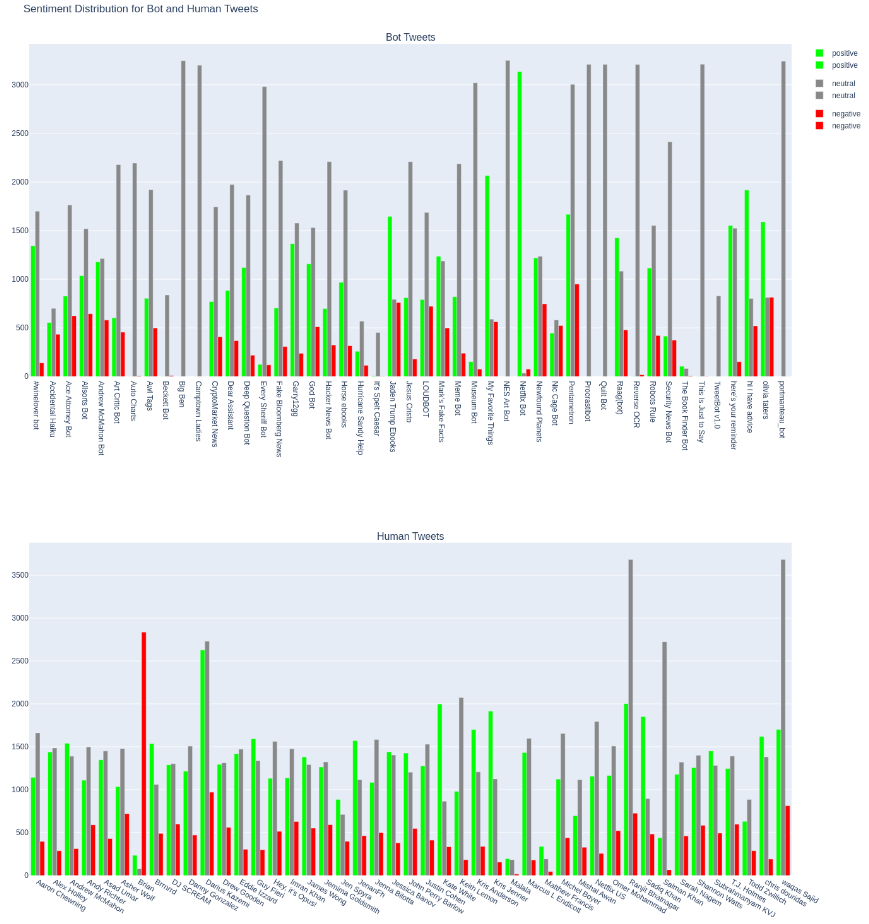

FIGURE 3. *The Tweets Sentiment*

Research Methodology

Based on the analysis of the characteristics of the experimental data, this project aims to study the effectiveness of text classification models in identifying Bot users within the Twitter environment. In this task, there are two classification categories: determining whether a tweet is generated by a Bot (label 1) or a human (label 0).

After constructing the text classification model, this experiment will explore two types of Bot identification processes. The first process focuses on identifying Bots at the tweet level, while the second process examines user-level Bot identification. Below, we will sequentially outline the specific methods and procedures employed in this experiment.

Introduction to Text Classification Tasks

Text classification is a fundamental task in natural language processing aimed at automatically assigning a given text to predefined categories. A typical text classification task involves several key steps:

1. **Text Data Processing:** This step involves cleaning and preparing the text data. It includes removing irrelevant characters, punctuation, converting to lowercase, and tokenization.
2. **Feature Extraction:** In this stage, text tokens are transformed into a format suitable for model processing. Common feature extraction methods include Bag of Words and TF-IDF (Term Frequency-Inverse Document Frequency).
3. **Model Building and Training:** Here, an appropriate machine learning model is selected for classification. Options include Naive Bayes, Support Vector Machines (SVM), Logistic Regression, and Random Forests.
4. **Model Evaluation:** The final step involves assessing the model's performance using relevant evaluation metrics. Common metrics for classification tasks include accuracy, precision, recall, and the F1 score.

Introduction to TF-IDF Features

In this task, we will utilize TF-IDF (Term Frequency-Inverse Document Frequency) features, a widely-used method in text mining and information retrieval. This method employs the TF-IDF score to evaluate the significance of a word within a document relative to a collection of documents. The calculation of the TF-IDF score involves the following steps:

1. **Calculating Term Frequency (TF)**: TF measures how frequently a word appears in a document, specifically the number of times a particular word occurs within that document.
2. **Calculating Inverse Document Frequency (IDF)**: IDF assesses the importance of a word across the entire corpus. It is calculated by taking the total number of documents in the corpus and dividing it by the number of documents that contain the specific word. The goal of the IDF value is to reduce the weight of common words that carry little informative value (e.g., "is", "are").
3. **Calculating the TF-IDF Value**: The final TF-IDF score is computed as follows:

$$\text{TF-IDF} = \text{TF} \times \text{IDF}$$

This method helps us highlight words that are important in specific documents, thereby enhancing the quality of our text analysis tasks.

Introduction to Classification Models

In this experiment, three classification models are employed to determine whether a tweet was sent by a bot user. The models used are Naive Bayes (Bayes 1968), Logistic Regression (LaValley 2008), and Random Forest (Breiman 2001). Below is an overview of each model and its underlying principles.

Naive Bayes Model

Naive Bayes is a classification method based on Bayes' theorem. It assumes that the influence of each feature on the result is independent of the others. While this assumption does not always hold in real-world scenarios, it simplifies the calculations, making them efficient. Naive Bayes is especially suited for text classification tasks, such as spam detection, as it can quickly train and predict on large datasets.

When using Naive Bayes for classification, we compare the posterior probabilities of different categories. The calculation formula is as follows:

$$P(C_k \mid X) \propto P(C_k) \cdot \prod_{i=1}^{n} P(x_i \mid C_k) \tag{1}$$

By identifying the category C_k that maximizes the above expression, the classification task is accomplished.

Logistic Regression Model

Logistic regression is a linear model used for binary classification. Despite its name containing the word "regression," it is primarily used for classification purposes. It maps a linear combination of input feature values to a probability using an S-shaped (sigmoid) function, allowing for effective classification. Logistic regression is straightforward and effectively handles the relationship between two classes, commonly applied in prediction tasks across finance, healthcare, and other fields.

Random Forest Model

Random Forest is an ensemble model composed of multiple decision trees. It constructs many decision trees, each trained on a subset of the data, and then lets these trees "vote" to decide the final classification. This approach enhances prediction accuracy and helps control overfitting. Random Forest is a powerful model suitable for both classification and regression tasks, and due to its flexibility and robustness, it is widely used in various data analysis applications.

Bot Identification Process Based on Tweets

In this experiment, we explore the first approach for identifying Bot accounts, which is based on the characteristics of individual tweets. Each tweet is treated as a standalone document. If we determine that a tweet was sent by a Bot, it leads us to conclude that the account responsible for sending that tweet is indeed a Bot account. The detailed processing workflow is illustrated in Figure (4).

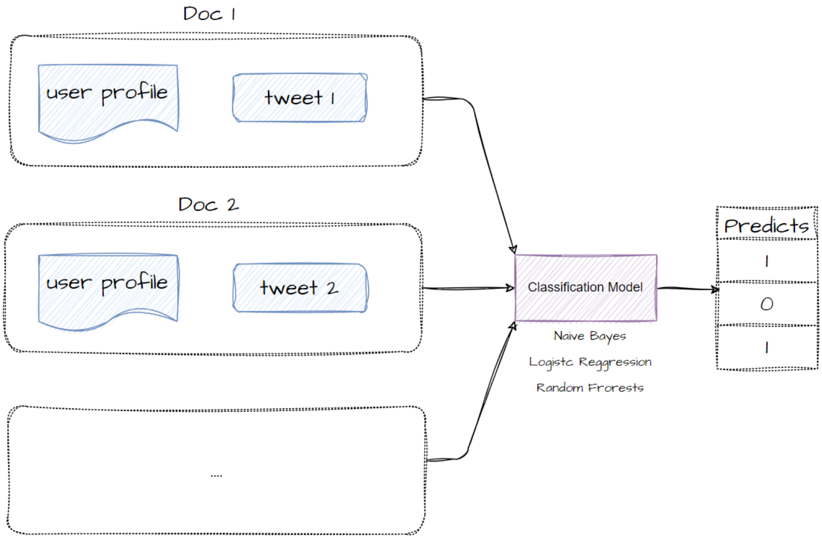

FIGURE 4. *Tweet Based Bot Detection Process*

User-Dimension Based Bot Detection Process

The bot detection process based on tweet dimensions has several drawbacks, including:

1) Not every tweet from bot users exhibits characteristics of bots, which may lead to misjudgment. 2) Evaluating individual tweets fails to consider the user's behavior from a holistic perspective, hindering a comprehensive assessment based on these behaviors.

To enhance the robustness of bot detection, we propose a user-dimension based bot detection approach, the detailed process of which is illustrated in Figure 5.

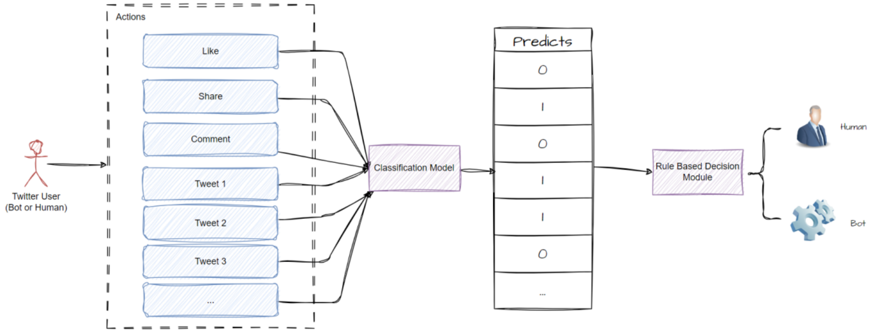

FIGURE 5. *User Based Bot Detection Process*

This process encompasses the following features:

1. It is essential to consider the overall behavior of a user comprehensively. 2. A Twitter user may exhibit a range of behaviors, including:

- Posting tweets
- Engaging in interactions such as liking, retweeting, and saving posts.

The process consists of two core submodules: the first is a classification model, which can directly reuse the models employed in the tweet-dimension based bot detection process. The second is the Rule-Based Decision Module, which integrates the detection results of individual behaviors to yield a final judgment. In this study, we adopt the ratio of individual behaviors identified as bot-like as the determining criterion. If the proportion of a user's actions identified as bots surpasses a certain threshold, that user will be classified as a bot user, as illustrated in Figure 6.

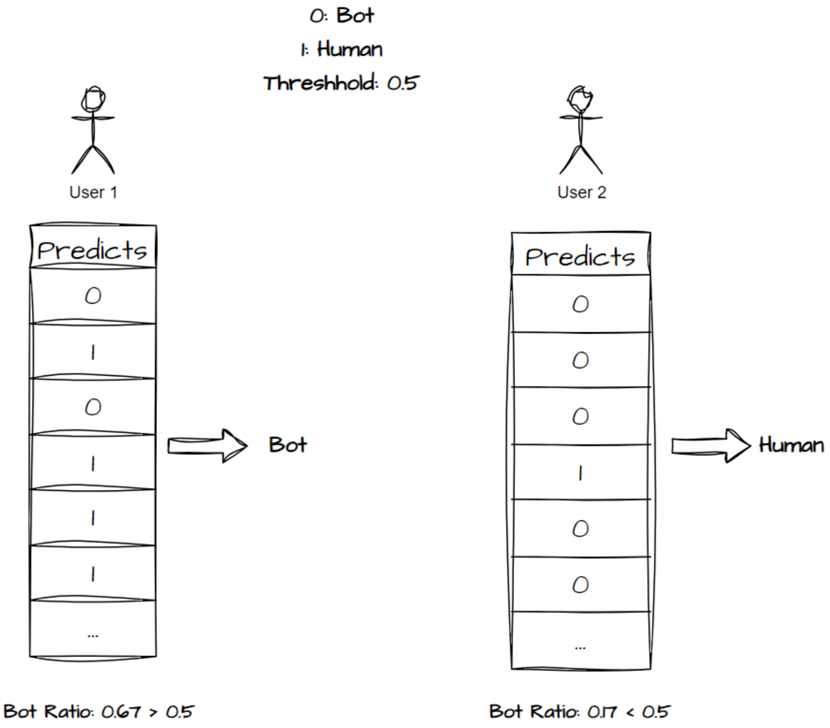

FIGURE 6. *Rule-Based Decision Module*

Experiment on Disguise Strategies

In real-world scenarios, creators of Bot accounts often employ certain disguise strategies to evade detection by monitoring systems. (Ferrara 2018) Common disguise strategies include:

1. **Mimicking Normal Posting Behavior**: Bot accounts typically simulate the posting style and frequency of ordinary users. For example, they may regularly publish relatively normal content and maintain reasonable time intervals between posts in order to avoid being identified as exhibiting abnormal activity. This content may mimic trending topics or popular trends, further helping them blend into the community.
2. **Acquiring Inactive Real Accounts Through Purchase or Hacking**: Some creators resort to purchasing forgotten or inactive real accounts from the black market, which they then utilize for Bot operations. This method effectively reduces the risk of detection since such accounts usually possess a certain history

and follower base, making them less likely to be immediately identified.

3. **Social Engineering**: By masquerading as ordinary users and interacting with others, Bots can gradually establish trust. This includes actions such as liking, commenting, and sharing, thereby pretending to be an active community member, which makes it more challenging to uncover their Bot identity.

In this task, we aim to incorporate some disguise techniques into the test dataset to verify the robustness of the detection system when faced with these disguise techniques. Specifically, this experiment employs the following procedure to simulate the disguise strategies of Bot accounts:

1. In the posts of Bot accounts, normal user posting content is mixed in at uniform time intervals according to a certain proportion. This involves concatenating a portion of historical tweet data from normal accounts with new tweet data from Bot accounts to construct new fake accounts, thereby simulating those accounts that were improperly acquired for the construction of Bot disguises.
2. A time period is used as a window for the determination of Bot accounts. If an account exhibits a higher proportion of Bot behavior within a certain time period, it is classified as a Bot account. The specific process is illustrated in Figure 7.

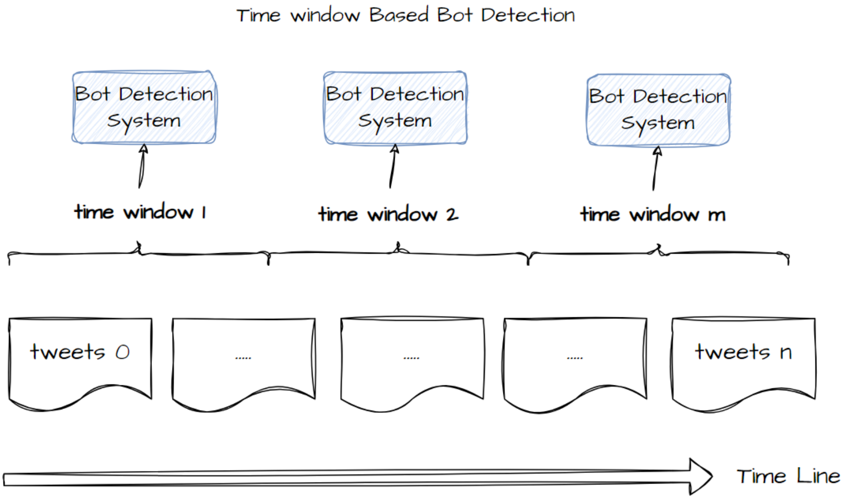

FIGURE 7. *Time-Window Based Process*

Experimental Comparison

Evaluation Metrics

This experiment uses accuracy as the evaluation metric, validating the algorithmic strategies by splitting the dataset into an 80:20 ratio.

Comparison of Different Classification Models

The performance comparison of different machine learning models on the dataset in this scenario is illustrated in Figure 8. From the experimental results, it can be seen that the logistic regression model exhibits the best performance in terms of accuracy, achieving 66.8%. This indicates that logistic regression is capable of effectively capturing the features within the data, making it suitable for the classification task in this dataset. In contrast, the accuracy of the random forest model is 61.9%, slightly lower than that of the logistic regression model. Although random forests typically demonstrate greater robustness in handling complex data, their performance did not exceed that of logistic regression on this specific dataset. This may be related to feature selection or the distribution of features within the dataset. As for the naive Bayes model, it achieved an accuracy of 60.0%, the lowest among all models. This suggests that the naive Bayes model may be constrained by its fundamental assumptions on this dataset, particularly when the independence assumption between features is not satisfied.

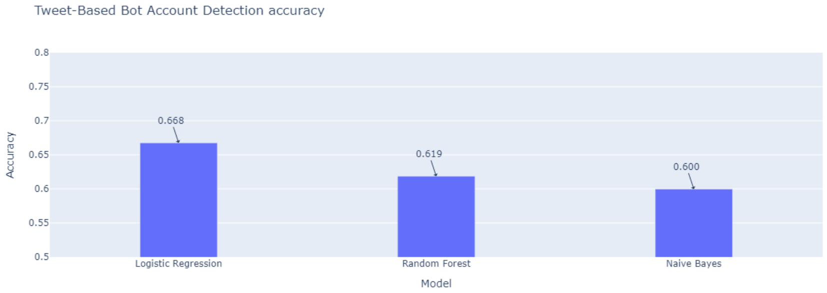

FIGURE 8. *Tweet-Based Process Experiments*

User-Dimension Based Recognition Effectiveness

The performance of three models at different thresholds in the user-dimension based identification process is illustrated in Figure 9. From the experimental results, it can be observed that when the threshold is set between 0.7 and 0.8, the bot detection based on the user dimension achieves the best performance. Among the models evaluated, Logistic Regression remains the most effective. As showing in Figure 10 Utilizing this model leads to an overall improvement of approximately 20 percentage points in recognition performance compared to the tweet-based strategy, with an accuracy metric reaching 86.8%.

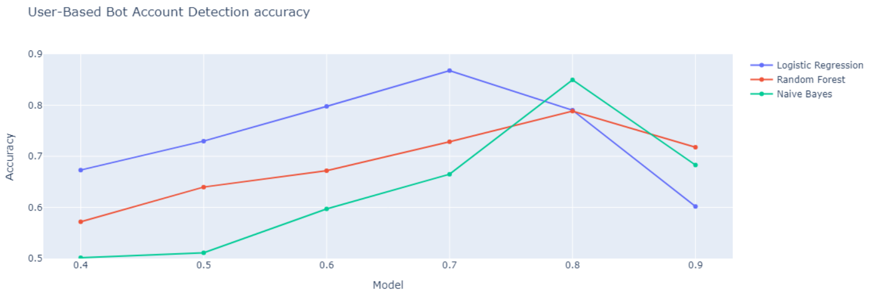

FIGURE 9. *User-Based Process Experiments*

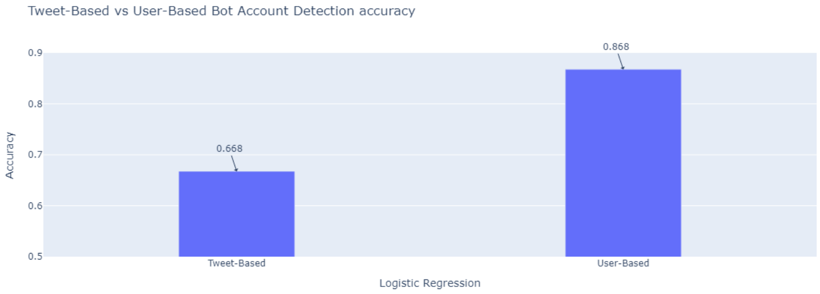

FIGURE 10. *"Tweet-Base Process VS User-Based Process" Experiments*

Experiment on the Recognition Effectiveness of Disguise Technologies

Comparative Effectiveness Against Disguise Technology 1

The experimental results for Disguise Technology 1 are illustrated in Figure **??**. It can be observed that when tweets sent by Bot accounts randomly include 25% of tweets from human accounts, both strategies exhibit a certain degree of decline in effectiveness. However, as long as an appropriate threshold is selected, the user-based recognition strategy can still maintain stability at around 82.5%. The decline magnitude for the two strategies is −6.13% and −4.95%, respectively, indicating that the decline in effectiveness of the user-based recognition strategy is smaller, demonstrating higher robustness.

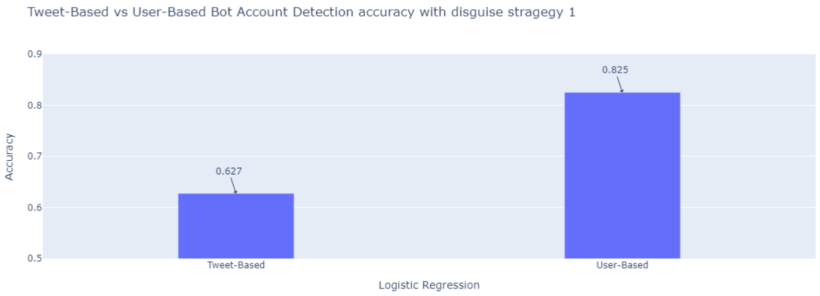

FIGURE 11. *"Tweet-Base Process VS User-Based Process" Disguise Technology 1*

Comparative Effectiveness Against Disguise Technology 2

The experimental results for Disguise Technology 2 are illustrated in Figure 12. It can be observed that in cases where accounts that were once normal human accounts have been subsequently transformed into Bot accounts, both tweet-based and user-based methods experience a significant drop in effectiveness. However, by incorporating a time-window recognition strategy based on the user dimension, this method exhibits excellent robustness against this type of disguise technology. As showing in Figure 13 When the decision threshold is set between 0.7 and 0.8, the overall recognition performance remains almost consistent with conditions where disguises were not applied.

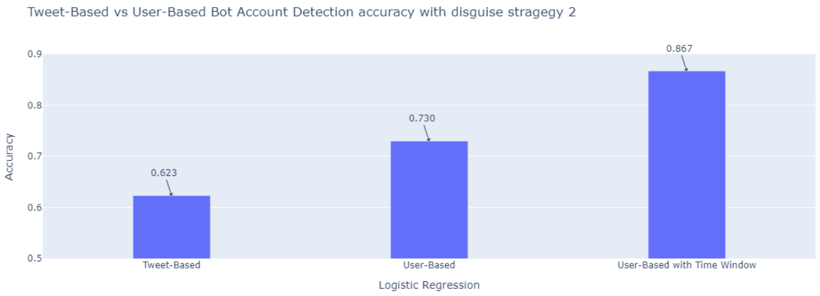

FIGURE 12. *Tweet-Base Process VS User-Based Process VS User-Based Time-Window Process*

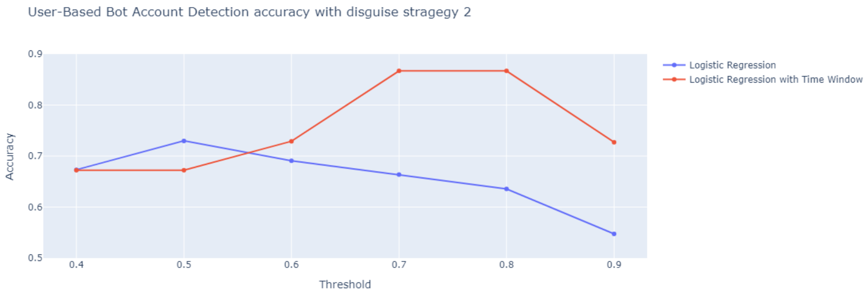

FIGURE 13. *User-Based Process Experiments On Disguise Technology 2*

Summary and Outlook

This project primarily investigates the application of machine learning algorithms in the identification of Bot accounts, as well as methods to enhance the robustness of identification systems.

Firstly, the project compares the performance of different machine learning models on Twitter data, while also experimenting with the effectiveness of Twitter Bot account identification based on user-level dimensions. It contrasts the performance of user-level processes with that of tweet-level processes, thereby demonstrating the significance of comprehensively considering user behavior information in the task of Bot account detection.

In addition, in response to common Bot account masquerading techniques, this project also experiments with the robustness of these strategies under different masquerading methods. By incorporating a time-window-based Bot account detection approach, the user-level detection system exhibits relatively stable performance across various masquerading techniques.

In future research, in-depth analysis of user behavior could be pursued. Moreover, integrating various link information from social networks and attempting to further optimize Bot account detection performance and robustness based on graph results could be beneficial.

References

Alarfaj, Fawaz Khaled, Hassaan Ahmad, Hikmat Ullah Khan, Abdullah Mohammaed Alomair, Naif Almusallam, and Muzamil Ahmed. 2023. Twitter bot detection using diverse content features and applying machine learning algorithms. *Sustainability* 15 (8):6662.

Aljabri, Malak, Rachid Zagrouba, Afrah Shaahid, Fatima Alnasser, Asalah Saleh, and Dorieh M Alomari. 2023. Machine learning-based social media bot detection: a comprehensive literature review. *Social Network Analysis and Mining* 13 (1):20.

Bayes, Thomas. 1968. Naive bayes classifier. *Article Sources and Contributors,* 1–9.

Breiman, Leo. 2001. Random forests. *Machine learning* 45:5–32.

Ferrara, Emilio. 2018. Measuring social spam and the effect of bots on information diffusion in social media. *Complex spreading phenomena in social systems: Influence and contagion in real-world social networks,* 229–255.

LaValley, Michael P. 2008. Logistic regression. *Circulation* 117 (18):2395–2399.