electronics

MDPI

Article

Detecting Fake Accounts on Social Media Portals—The X Portal Case Study

Weronika Dracewicz and Mariusz Sepczuk *

Faculty of Electronics and Information Technology, Warsaw University of Technology, 00-665 Warsaw, Poland; weronika.dracewicz.stud@pw.edu.pl
* Correspondence: mariusz.sepczuk@pw.edu.pl

Abstract: Today, social media are an integral part of everyone's life. In addition to their traditional uses of creating and maintaining relationships, they are also used to exchange views and all kinds of content. With the development of these media, they have become the target of various attacks. In particular, the existence of fake accounts on social networks can lead to many types of abuse, such as phishing or disinformation, which is a big challenge nowadays. In this work, we present a solution for detecting fake accounts on the X portal (formerly Twitter). The main goal behind the developed solution was to use images of X portal accounts and perform image classification using machine learning. As a result, it was possible to detect real and fake accounts and indicate the type of a particular account. The created solution was trained and tested on an adequately prepared dataset containing 15,000 generated accounts and real X portal accounts. The CNN model performing with accuracy above 92% and manual test results allow us to conclude that the proposed solution can be used to detect false accounts on the X portal.

Keywords: fake account detection; Twitter (X); machine learning; image classification

Citation: Dracewicz, W.; Sepczuk, M. Detecting Fake Accounts on Social Media Portals—The X Portal Case Study. *Electronics* **2024**, *13*, 2542. https://doi.org/10.3390/electronics13132542

Academic Editors: Abdussalam Elhanashi and Pierpaolo Dini

Received: 25 May 2024
Revised: 20 June 2024
Accepted: 24 June 2024
Published: 28 June 2024

1. Introduction

Social media have become an integral part of our lives. The findings in [1] stated that 59.4% of the population (4.76 billion) were active social media users as of 2023, a number that is continuously growing, with a 3% increase since 2022. Social media allow users to create and maintain relationships with others and are also a space for users to consume content or express themselves. Social media have provided us with not only entertainment for years but also, for many users, the primary source of information about the world, news, trends, or events. A social platform's role is also to provide entertainment and exciting content.

Companies also use social media as a tool to promote their products and services. A company can use social media platforms to promote its brand, products, and services by publishing photos, videos, and posts.

Among the many advantages of social media, there are also disadvantages. Particularly significant are those related to cybersecurity [2–4]. Criminals can easily create a false identity on social media and target potential victims. User activity provides a lot of valuable information, such as biographical data, material assets, needs, or personality traits. This information can be used for crimes such as the following:

- Matrimonial fraud;
- Phishing (impersonating another person or institution to obtain essential data);
- Hacking (e.g., breaking into a user's computer and taking control of it);
- Cyberstalking (online harassment).

Another disadvantage of social platforms is the speed at which people spread false information. The purpose of incorrect information (fake news), which has become extremely popular, is to mislead and harm people.

That is why it is essential to check everything from several sources and to be able to distinguish real news from fake news. However, we often do not have enough time to verify all the information we have acquired. Undoubtedly, one of the elements of such a check is to determine the authenticity of the account from which each piece of content comes. This paper focuses on a solution that detects fake accounts on the X platform based on its appearance. The main contributions of this paper are summarized as follows:

- Present a distinctive visual-based approach to account classification;
- Create an image dataset of platform X accounts;
- Validate the created dataset;
- Test the detection of the authenticity of an X portal account by using the selected machine learning model.

It should be noted that despite the many solutions related to image classification, there are few that use this approach in social networks, especially on the X platform. The text-based method is a more popular approach and performs classification based on separately extracted data, while the proposed solution treats the image as a whole. The rest of this paper is organized as follows: Section 2 contains a brief review of related work in the field of fake account detection in social networks. Section 3 describes in detail the construction of the used dataset and its fundamental characteristics. Section 4 refers to initial experiments on the created dataset and shows their results. The article ends with a discussion of the results (Section 5) and conclusions (Section 6).

2. Related Works

The statement that social media have become an integral part of our lives in recent years is not a bit exaggerated. Today, many people find it hard to imagine a day without browsing Facebook, Instagram, or Twitter. This can also be seen in the statistics of the overall use of these media between February 2023 and February 2024 [5]: Facebook (65.46%), Instagram (9.92%), Twitter (6.7%), and LinkedIn (less than 1%). Each of these portals takes a different approach to detecting fake accounts. The literature has many examples of concepts for classification using ML. In general, fake profile identification using ML is not a new idea, but new papers that show another aspect of such a detection method are still being written.

In [6], the authors focused on detecting fraudulent accounts created by humans. They used Random Forest to identify if the user's profile was authentic with TF-IDF vectorization (a numerical representation of words utterly dependent on the nature and number of documents under consideration). The paper [7] describes how to detect fake Instagram accounts based on the textual data from this OSN (online social network). ML algorithms such as K-NN, Logistic Regression (LR), and decision tree (DT) were utilized to identify these accounts. In the paper ref. [8], the authors present a supervised model for detecting fake accounts based on machine learning to deal with the spread of fake content, which could be a valuable tool for controlling the excesses of online social crime. Another text-based approach is described in [9]. Yet another text-based solution was included by Bhattacharyya and Kulkarni in the article [10]. They explained how to detect fraudulent accounts and classify them into four categories: genuine accounts, social spambots, traditional spambots, and fake followers. Goyal et al. described in [11] a new solution that combined textual, visual, and network-based features to identify the different characteristics of fake accounts. The proposed approach used a deep neural network incorporating CNNs, LSTM, and graph GCNs to analyze these features.

The paper [12] applies to a fake profile detection model that uses emotion-based features to recognize a real or fake account. The approach was trained on 12 emotion-based attributes (both positive and negative), and the experiments were conducted on Facebook posts. On the other hand, the article [13] describes a method for detecting fake accounts by using a set of 17 features which are essential to distinguishing unreal users on Facebook from real users. Another solution for detecting fake accounts on Facebook can be found in [14]. The authors describe Intergo as a scalable defense system that uses a meaningful

user ranking scheme. First, the solution predicts victim accounts (real accounts connected with fake accounts) from user-level activities. Based on that, a weighted graph and a list of potential fake accounts are created. In the paper [15], considerations on mitigating the problem of stealing a real account and replacing it with a fake one are presented.

The author of the paper [16] depicted a method for identifying Instagram fake accounts. The proposed method utilizes a gathered dataset used in the bagging classifier to classify forged accounts. Moreover, the created solution was compared to several well-known machine learning classifiers in terms of classification accuracy to evaluate the method's effectiveness better. A similar approach can be found in [17]. The document [18] includes classifications of impersonator categories that can exist on Instagram. First, the authors, using crawlers, collected the activities of famous politicians during a defined period. Then, they experimented with data from these profiles and built a model that could recognize fake accounts. Another example of detecting counterfeit profiles on Instagram is presented in [19]. The paper applies to a solution of fake account identification using supervised learning machine algorithms. The detected bogus profiles were stored to help the authorities take essential action against fraudulent social media accounts.

In the context of LinkedIn, we can also find solutions for detecting fake accounts [20,21]. In [20], the authors determined the minimal set of profile data necessary to specify if a profile is fake and describe an idea of a data mining approach that can be used for such identification. The results allowed them to conclude that the approach gave roughly similar results to those based on large sets of features used for detection. In contrast, in the document [21], an approach to finding groups of fake profiles created by the same actor was depicted. The designed solution used a supervised machine learning algorithm to classify an entire profile cluster as real or forged. This can be achieved based on user-generated text, such as email address, name, or residence.

Yet another approach was used for the X portal (formerly Twitter). The author in [22] used a multi-objective hybrid feature selection idea to better classify fake profiles. The selection of features was performed by using the Minimum Redundancy–Maximum Relevance algorithm (mRMR). The paper [23] employed a Chrome extension that detects fake profiles on Twitter by analyzing the different features. This approach uses, for account identification, both Random Forest and bagging methods. An interesting case is described in [24]. The paper contains a spam recognition artificial intelligence method for Twitter social networks. The model in the proposed solution was built by using a vector support machine, a neural artificial network, and Random Forest algorithms. Finally, the document [25] considered the detection of fake accounts generated by humans, as opposed to these created by bots. For such a distinction, a dedicated set of features was selected and applied to different types of supervised machine learning models.

From a preliminary review of the literature, it can be noted that not many solutions detect fake accounts based on images (see Table 1)—they are usually datasets of specific text data associated with a particular account. In our proposed solution, detection will be based on pictures of the profiles of X accounts. A detailed description of the solution can be found in the following sections.

Table 1. Comparison of methods for detecting fake accounts on X portal.

No.	Paper	Problem	Approach
1	[10]	Fake X account detection	Text-based solution
2	[11]	Fake X, Instagram, and Facebook account detection	Combination of text, visual, and network factors
3	[22]	Fake X account detection	Text-based solution
4	[23]	Fake X account detection	Text-based solution
5	[24]	Fake X and Facebook account detection	Text-based solution
6	[25]	Fake X human-created account detection	Text-based solution
7	This paper	Fake X account detection	Image-based

3. Creating a Dataset of Twitter Accounts

In this case study, we focused on detecting fake accounts on the X portal. The X network portal was selected due to persisting areas for improvement in ensuring the privacy and authenticity of its users while accessing the portal [26]. To train the image classification model, we needed to create our dataset consisting of figures of X accounts, since the existing and publicly available datasets did not meet our needs, as none provided graphical data nor had all the necessary characteristics to define particular classes of accounts. Below, we present the process of creating our dataset, from defining features to implementing a cloned view of the original X social network and figure generation process. This procedure could be helpful for future research and similar studies for other portals and types of accounts.

3.1. Definition of Various Types of Accounts

An important consideration when dealing with the issue of detecting fake accounts is to perform an initial security analysis of social networks and characterize the types of users that appear. In this subsection, the identified types of profiles are described, along with their features and the intentions of creation.

Profiles on social networks can vary according to their features and objectives. Due to their widespread use and accessibility, social platforms are used for countless purposes. The general public can exchange messages, photos, videos, and blog posts and communicate with people. Nevertheless, being aware of the Internet's unethical side is essential to quickly recognizing and understanding its intentions. This subsection lists the types of users we have learned about and identifies their impact on online security.

According to the authors of the article [27], who studied a group of more than 100,000 users of the Twitter platform and classified their roles based on the followed/followers ratio count, Twitter users fall into three groups: (1) broadcasters, (2) acquaintances, and (3) so-called miscreants (e.g., spammers). The nature of the interconnections among users in social networks and the previously mentioned relation are presented with a scatterplot in [27].

Broadcasters. This type of user has a much larger number of followers than they are following themselves. Many of these users represent recognizable individuals, online stores, radio stations, magazines, and other large organizations that use Twitter to promote their offerings and products and interact directly with their target audiences.

Acquaintances. The second group of users, referred to as acquaintances, tend to show mutuality in their relationships, a common characteristic of online social networks. In general, these are real users interacting with the social network for its intended function to connect with friends and family or to follow people from the broadcast group.

Miscreants. The common feature of the users included in the third unique group is that they follow a much larger number of people than they have followers. This behavior is typical of spammers or people who actively promote their beliefs. While this is one of the most suspicious groups, it does not exclude the possibility that there are genuine users with many interests or preferences.

The previously referred classification of users is a fundamental knowledge base of the accounts encountered. However, we need a complete understanding of the behavior of fake accounts. Thus, this also proves that a classification based on a single account feature cannot be a sufficient basis for further research.

A deeper analysis must consider several aspects to identify the types of users correctly. The authors of the publication [28] presented the results of a similar analysis on the classification of people using Twitter into three groups: (1) human, (2) bot, and (3) cyborg—a group that includes humans assisted by bots and bots assisted by humans. This analysis included the tweets' content, the occurring URLs, the tweeting devices, the user's profile, and the number of followers and friends.

Human. A user is considered and labelled a human if their profile contains authentic, meaningful, and concrete content. In particular, real users usually write down what they

do or how they feel about something that appears on Twitter. Thus, they use it as a microblogging platform to self-express while interacting with friends. Concreteness means that the tweet's content is presented in relatively straightforward words with awareness, e.g., the answer to the question is relevant and directly addresses its subject.

Bot. Users demonstrating a lack of human-intelligent or original content are assigned to the bot group. Examples of such behavior include endlessly forwarding (retweeting) other people's posts and posting or advertising tweets with identical content. Frequently, this results in the propagation of unethical and false information, broadly published by using fake profiles. Excessive automation and the abundant presence of spam or malicious URLs in a user's profile also expose the account's association with such a group. Attackers use malicious bots to send spam and phishing messages, spread malware, host Command and Control (C&C) channels, and launch other illegal operations. In the case of external URL links, the main warning is that there is no connection between the link and the post's content. Audited by eye-catching text, users may click on links and be redirected to harmful sites. The last and least obvious of the characteristics is aggressive behavior toward other users, aiming to attract more attention (e.g., following and massively unfollowing quickly). Bots randomly add other users as friends, aiming to reach a large audience. In such a way, spam tweets posted by the bots are displayed on other users' recommendation pages.

Cyborg. According to the definition of this group, cyborgs, upon analyzing the behavior, include users where it is possible to find indications of both human and bot roles. For example, a typical cyborg account may contain different types of tweets. Many will have content with human-like intelligence and originality, while the rest will be automatically published. That represents a usage model whereby a human uses an account occasionally while applying automated approaches most of the time.

3.2. Feature Engineering to Generate a Dataset

The essential step required to create a detection solution for fake profiles is to define a set of features, determining whether an account belongs to one of the groups described earlier. The mentioned account characteristics are discussed and described in detail in this subsection.

The paper [26] grouped the most critical studies on detecting fake Twitter accounts according to the users' profile characteristics that were taken into account. Such a list of features was considered while defining and engineering types of accounts for the database. As mentioned, the dataset of profiles' images should combine text features and account-based data. For this reason, Table 2 contains features chosen among the characteristics presented in [26] to design the profiles.

Table 2. Selected features used to identify account type.

No.	Selected Feature	Description of Feature
1	Username	Unique identifier/name of user's account
2	Biography	Short introduction written by users about themselves, their achievements, expertise, and other important information
3	Profile photo (avatar)	One of the main features of accounts; it allows one to recognize a person by their appearance more quickly and easily
4	Header photo (banner)	In addition to the previous, Twitter introduced such photos to make the user's account more attractive
5	Date of creation	The date when the user created their account and became active on the network portal
6	Website	URL link that could be the user's website or profile on other platforms
7	Number of tweets (Twitter posts)	The essential feature for fake profile detection that allows for the determination of the level of user activity
8	Number of followers	Number of other accounts that are following the user
9	Following count	Number of other accounts that are being followed by the user's profile
10	Number of likes	An important feature indicating the number of profiles that liked the content created by the user
11	Number of views	Number of profiles that have seen the content created by the user, showing how wide their audience is
12	Number of retweets	Number of how many times the user's content was shared on both Twitter and other platforms
13	Number of replies	Number of comments on the user's posts

In addition to the listed features, verification was included, which indicates account authenticity granted by Twitter.

3.3. Types and Characteristics of Generated Accounts on X Portal

The research conducted focused on four classes of accounts that were defined by applying the previously mentioned characteristics and behavioral analysis: (1) bot, (2) cyborg, (3) real, and (4) verified. Table 3 presents profile features for the selected classes of accounts.

Table 3. Attributes for different classes of accounts.

<table>
<tr><th rowspan="2">Characteristics</th><th colspan="4">Classes of Accounts</th></tr>
<tr><th>Bot</th><th>Cyborg</th><th>Real</th><th>Verified</th></tr>
<tr><td>Profile photo</td><td>Blank
or
default (initials)</td><td rowspan="2">Blank or default (initials)
or
both or only profile</td><td rowspan="2">Blank or default (initials) + header
or
both or only profile photo</td><td>Yes</td></tr>
<tr><td>Header photo</td><td>No</td><td>Yes</td></tr>
<tr><td>Account description</td><td>No</td><td>No</td><td>Yes</td><td>Yes</td></tr>
<tr><td>Account website</td><td>No</td><td>No</td><td>Website URL
or
no website</td><td></td></tr>
<tr><td>Number of followers</td><td colspan="2">Low number of followers
or
no accounts following a given profile</td><td rowspan="2">Average</td><td>High</td></tr>
<tr><td>Number of followings</td><td colspan="2">High</td><td>Average</td></tr>
<tr><td>Date of creation</td><td colspan="2" rowspan="3">Large post No. + low interactions No. (close date)
or
no posts (former date of account creation)</td><td>former</td><td>Former</td></tr>
<tr><td>Number of posts</td><td>Average</td><td>High</td></tr>
<tr><td>Number of interactions</td><td>Average</td><td>High</td></tr>
<tr><td>Verification</td><td>No</td><td>No</td><td>No</td><td>Standard (blue icon)
or
business (yellow icon)
or
institutional (gray icon)</td></tr>
</table>

The outlined definition of bots, also shown in Figure 1a, allows us to consider most bot use cases. It considers the automation processes used to publish posts and the massive creation of empty accounts with no activity. The lack of any signs of human intelligence also marks them.

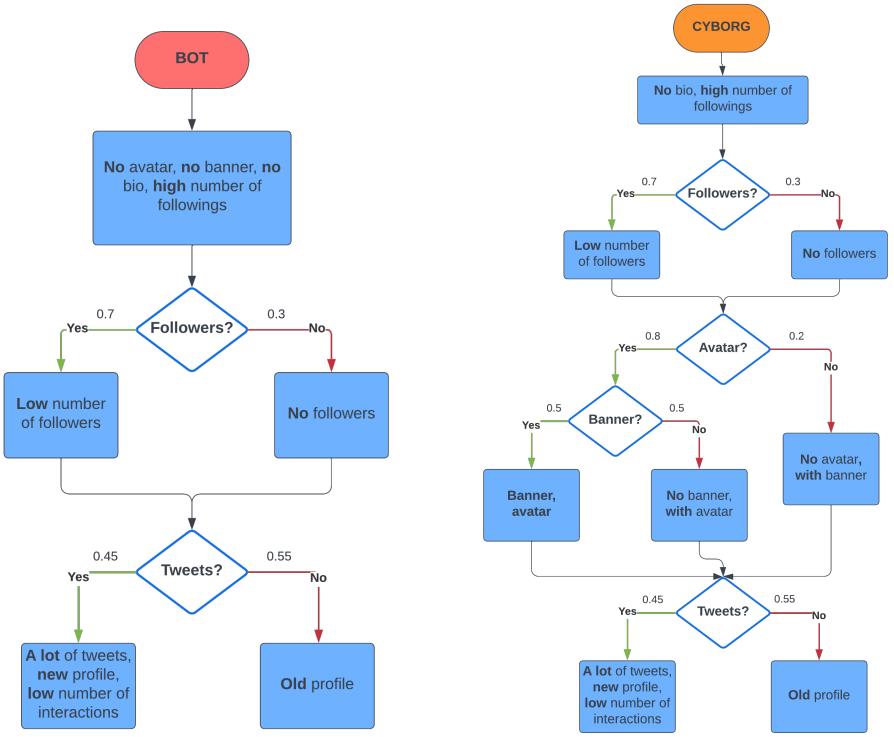
The defined cyborg profiles have similarities with accounts of bot nature. However, they are unique in their combination of human and automated behavior. The features designed to create this group are also illustrated in Figure 1b.

Generated by the outlined features, accounts of actual users cover most human use cases of social networks. People will look for mutuality by following other accounts and actively sharing their experiences through posts and replies. The designed class meets the definition of both more and less active people on Twitter, the detailed scheme of which is illustrated in Figure 2a.

The social network checks the content and authenticity of the profiles belonging to the verified class when granting verification. Thus, they represent actual and well-known users who can interact with each other and have many followers. In addition, Figure 2b schematically depicts the previously discussed features of verified profiles.

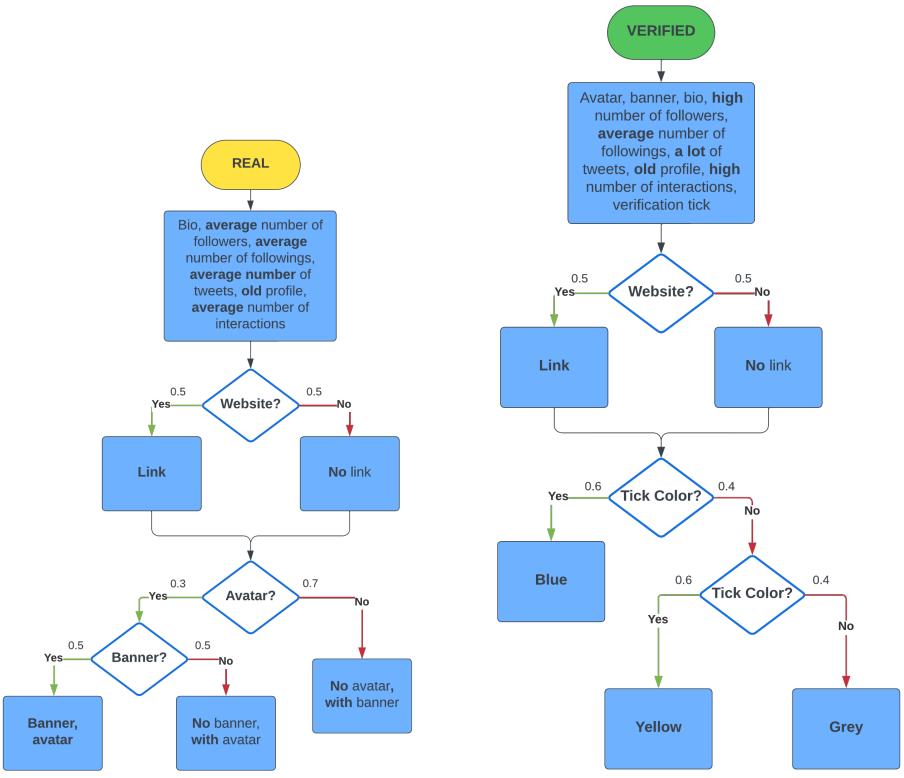
The defined features of each profile type and the generating principles for a set of images were appropriately adapted to the current appearance of the X social portal. This means the proposed and designed schemes do not represent the final definition of accounts' behavior. Therefore, they should be updated continuously to keep up with the dynamically evolving social platform. It is essential to be aware that considering all the

changes occurring in the social network when generating the dataset will allow us to train a more efficient model, which, on the other hand, leads to more effective predictions.



(**a**) Bot class (**b**) Cyborg class

Figure 1. Feature definition scheme for profiles of (**a**) bot and (**b**) cyborg classes.



(**a**) Real class (**b**) Verified class

Figure 2. Feature definition scheme for profiles of (**a**) real and (**b**) verified classes.

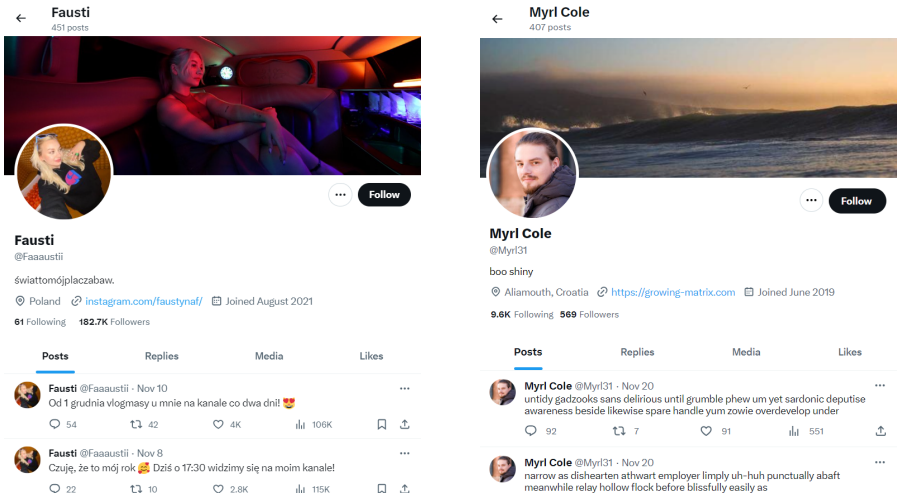
3.4. Generation and Presentation of Accounts' Images

In order to generate a dataset, an additional website was implemented that was a mock-up view of the X social network. Following the styles of the original homepage of a user profile, we projected a primary view that allowed us to substitute data representing each class of accounts.

For this aim, standard web technologies and tools were used, such as TypeScript—to write code for a web application that is a replication of X's web view; React [29]—which was used to create graphical interfaces for a web application; and Playwright [30], which is an open-source automation library for browser testing, was used to take screenshots containing user profiles.

Additionally, the data that were used to create profiles for the dataset came from the Faker.js library [31]—used for profile pictures, first names, last names, locations, bio, links, all numeric data and text statements generation, as well as the open API Lorem picsum [32], here was used to generate profile banners.

A comparison of the web views of user profiles in the original portal and the implemented environment is shown in Figure 3a,b.



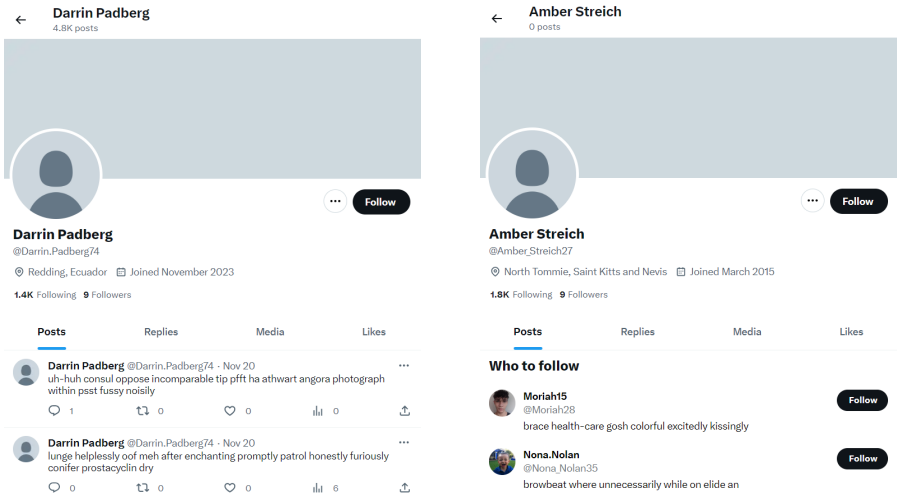
(**a**) Original web view (**b**) Implemented web view

Figure 3. Comparison of web views of (**a**) original and (**b**) implemented X profiles.

As mentioned, the website created allowed us to substitute the randomly generated classes' features defined previously dynamically. The following demonstration shows how the feature engineering performed on the dataset was applied and translated into the appearance of all types of accounts.

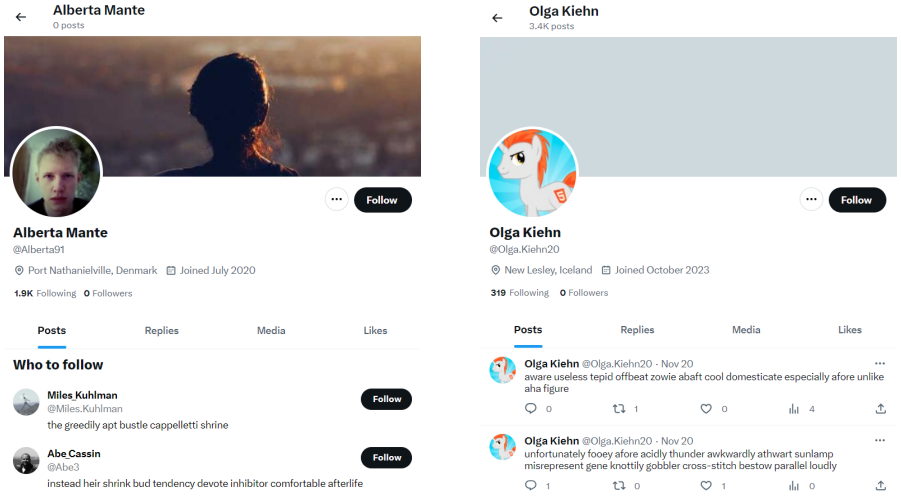
Figure 4 shows some examples of generated bot profiles. Their main feature is the lack of photos, additional information about the user, and a low level of activity and interaction. The first use case in Figure 4a shows an account recently created. However, the large number of posts and following profiles suggest automated control of it. In contrast, the second use case, shown in Figure 4b, presents an old account with no posts, but with a large number of following users. This behavior indicates a bot account created some time ago that has not been detected or deleted yet.

On the other hand, Figure 5 shows profile instances of the cyborg group. Their characteristics are similar to those of bots while having signs of human maintenance. For example, the use cases in Figure 5a,b present the profiles that are a combination of the behaviors we discussed above in the case of bots with profile pictures or/and header images.



(**a**) First bot use case (**b**) Second bot use case

Figure 4. Web views of accounts designed as bots.



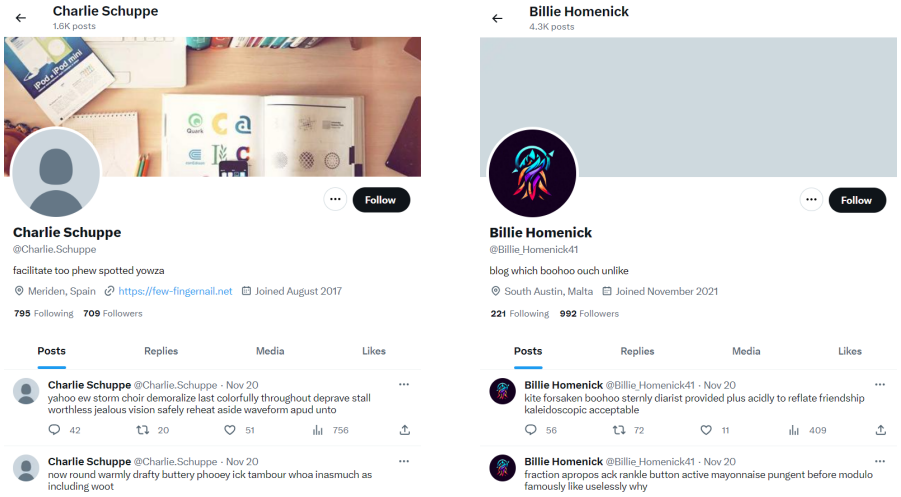
(**a**) First cyborg use case (**b**) Second cyborg use case

Figure 5. Web views of accounts designed as cyborgs.

The real profile class, an example of which is shown in Figure 6, represents users that tend to have profile photos and more detailed information about themselves on their social network account. Moreover, they have both a description of the profile and an adequate ratio of the number of followers/followings to the level of interaction and activity on the posted tweets. In Figure 6a, the first use case illustrates an account created some time ago but containing a banner, a URL website link, and many posts. The main reason profiles could be recognized as authentic is that an actual human using the X portal can achieve this level of characteristics. Additionally, an account with a profile picture that does not have a banner photo and a website can be classified as real due to an average user's lack of everyday use. Such an example of the profile can be seen as a second use case in Figure 6b.

Exemplary web views of accounts from the verified class, shown in Figure 7, are primarily defined as accounts of well-known and recognizable people. It is worth noting that X's verification feature significantly increases the likelihood of the account being legitimate. However, with time, this feature has become less reliable, as it is easier to gain verification without meeting all of its requirements or even purchase the verification sign. For this reason, verified accounts cannot be fully perceived as real, and analyzing them in the suggested way is essential. The main characteristics of this group remain: many followers, high-profile activity, and interaction. Despite this, truly verified users

usually have profile and header photos with an extended biography and information about themselves.



(**a**) First human use case (**b**) Second human use case

Figure 6. Web views of accounts designed as real users.

(**a**) First verified profile use case (**b**) Second verified profile use case (**c**) Third verified profile use case

Figure 7. Web views of accounts designed as verified profiles.

In the X social network, the type of verification is indicated by the color of the verification icon. Figure 7a, 7b and 7c show profiles with standard (blue icon), business (yellow/gold icon), and institutional (gray icon) types of verification, respectively.

In the manner described in this section, the substitution of data on the website was automated, and 11,000 unique profiles' images were created, where 10,000 (2500 per class) are data for training and 1000 (250 per class) for testing machine learning models. The dataset size was sufficient for machine learning, including the neural network models [33]. However, it could still be expanded for future experiments. It is worth noting that for the final evaluation of the detection tool, 4000 additional unique accounts were generated. Altogether, the developed dataset consisted of 15,000 screenshots of the accounts mentioned above with a resolution of 600 × 800 in PNG (Portable Network Graphic) format.

4. Experiments and Results

The dataset created, as discussed in the previous section, was used to train several of the most popular image classifiers. When the preferred ML model was selected and

optimized, we moved on to incorporating it into the working solution, with a friendly user interface. The process of creating the specific components of the tool for detecting fake profiles on the X platform, as well as test results, are discussed in this section.

4.1. Machine Learning Model Selection and Optimization

A supervised machine-learning approach was used to detect fake accounts on the X platform. In this subsection, we present an analysis and a comparison of three image classification methods to select the most efficient detection tool for further development.

The machine learning models that were studied are (1) a model based on the Convolutional Neural Network classifier [34], (2) a model based on the Random Forest classifier [35], and (3) a model based on the Naive Bayes classifier [36]. Each of the introduced models was trained on 10,000 images of X profiles and then tested on the additional 1000 test images (10% of the training data). The tests conducted allowed us to determine metrics such as accuracy (Equation (1)), precision (Equation (2)), recall (Equation (3)), and F1-score (Equation (4)).

$$\text{Accuracy} = \frac{Correct\ predictions}{All\ predictions} \tag{1}$$

$$\text{Precision}_{class\ A} = \frac{TP_{class\ A}}{TP_{class\ A} + FP_{class\ A}} \tag{2}$$

$$\text{Recall}_{class\ A} = \frac{TP_{class\ A}}{TP_{class\ A} + FN_{class\ A}} \tag{3}$$

$$\text{F1-score}_{class\ A} = 2 \cdot \frac{\text{Precision}_{class\ A} \cdot \text{Recall}_{class\ A}}{\text{Precision}_{class\ A} + \text{Recall}_{class\ A}} \tag{4}$$

where *class A* is bot, cyborg, real, or verified; TP: true positive; TN: true negative; FP: false positive; and FN: false negative.

The metrics resulting from testing the classifiers were averaged across all classes to obtain the final macro-averaged scores shown in Table 4.

Table 4. Metric comparison of the considered models of image classifiers.

Classifier	Accuracy	Avg. Precision	Avg. Recall	Avg. F1-Score
Convolutional Neural Network	96.5	96.59	96.40	96.49
Naive Bayes	87.27	89.1	87.29	86.89
Random Forest	80.26	85.35	80.25	79.31

As the tests showed, the Convolution Neural Network turned out to be the best model of the compared classifiers. Therefore, it was decided to use the CNN model for the machine learning component to detect fake profiles.

In the next step, the optimization of the selected classification model was carried out. For this purpose, we studied the impact of the network structure and the number of neurons used in its layers on the detection process's accuracy and final loss function. The training of each model variation lasted for 25 epochs; then, the accuracy and loss function values obtained on the training (8000 images) and validation (2000 images) datasets were compared. The loss function was defined as the cross-entropy between the labels and the predictions made by the model.

The selected neural network had a sequential model consisting of four convolutional layers with ReLU activation function and 3×3 kernel size, four max-pooling layers, a flattened layer, and two dense layers. The visualization of the architecture of the addressed neural network model is presented in Figure 8.

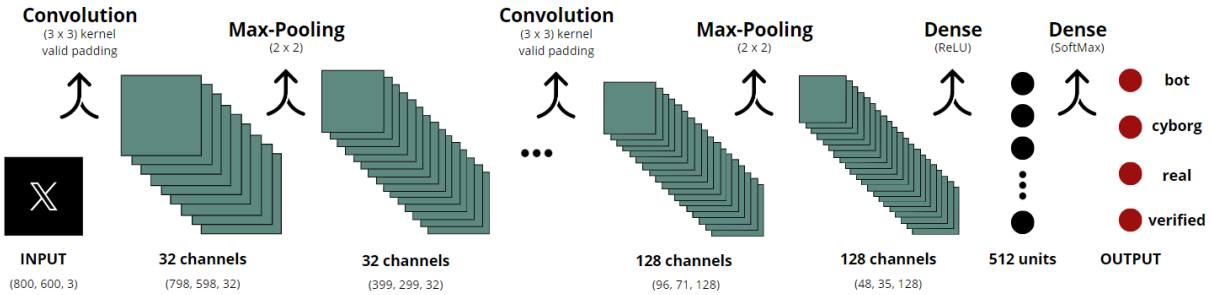
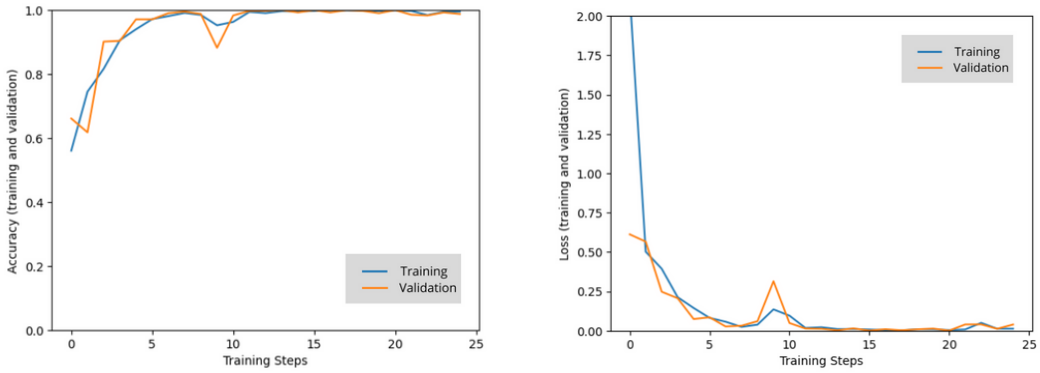


Figure 8. Design of CNN model's architecture.

Additionally, for the optimizer, we chose the Adam algorithm with a learning rate of 1×10^{-3}. The created model consisted of 110,343,876 learning parameters and weighed around 420.93 MB; the training process lasted about 2.5 h, and the final prediction time was 0.2 s.

The shape of the input data was set to (800, 600, 3), which means that the input images were 800 pixel high and 600 pixel wide and had three color channels (RGB—red, green, and blue). This corresponds to the size of screenshots taken of X's accounts, so there was no need for further data preprocessing.

Figure 9 presents graphs of the dependence of the accuracy (Figure 9a) and loss (Figure 9b) functions on the training period. The images were observed to show a correct increase in the accuracy value and a decrease in the loss value.



(**a**) Accuracy function (**b**) Loss function

Figure 9. Accuracy and loss function graphs for Convolutional Neural Network model.

Finally, the performance of the classification model was verified by using a test dataset (1000 images).

4.2. Detection of Fake Accounts

To check how our model will detect fake X portal accounts, a tool to ensure the most secure use of social networks was developed, enabling users to analyze any account in real time. Therefore, a web browser extension was considered the most convenient detection solution. This subsection will discuss the project assumptions and usage principles of the fake account detection tool in detail.

Among the commonly used web browsers, the Google Chrome browser based on the Chromium architecture was chosen to implement the detection solution. Moreover, an additional facade component was designed to communicate between the browser and the selected classification ML model. A pre-trained neural network classifier runs in a container environment, facilitating the developed tool's scalability, portability, and management.

We implemented WebSocket API [37] technology for communication, which made it possible to load the pre-trained model once within the first launch of the extension. In this way, obtaining the shortest and optimal waiting time for the returned detection model

response was possible. Afterwards, the web socket opened a two-way communication session between the browser extension and the facade component, allowing it to load the profile's image and return the prediction result to the users. Furthermore, when forwarding to the machine learning model input, we used `Base64` encoding to transfer the image from the client side and decode it on the facade side.

In terms of implementation, a web browser extension, which was given the name of "FakeDetector", making it available in the pop-up window format, made it possible to use the tool while browsing profiles on social networks. In this step, the user must navigate to the profile of interest, where a screenshot of the currently open tab is taken according to strictly set positions and sizes. Therefore, for the best performance, it is essential to correctly position the profile—ensuring all account information is visible—and set the screen scale to 100% to avoid unwanted scaling. The screenshots are taken by using the `chrome.tabs` API [38] and are then forwarded to the ML model input.

All of the detection components of the fake profile tool are illustrated in Figure 10.

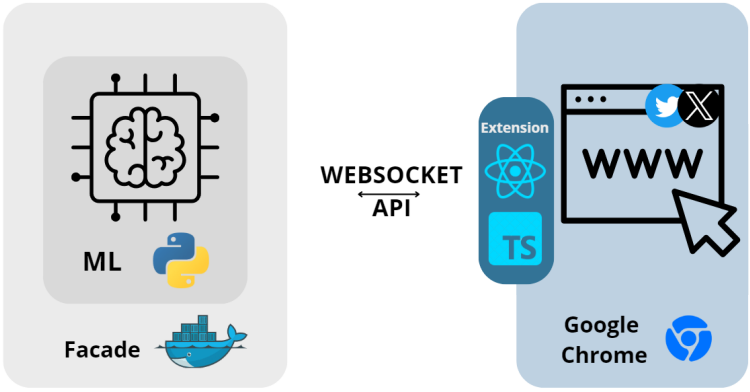
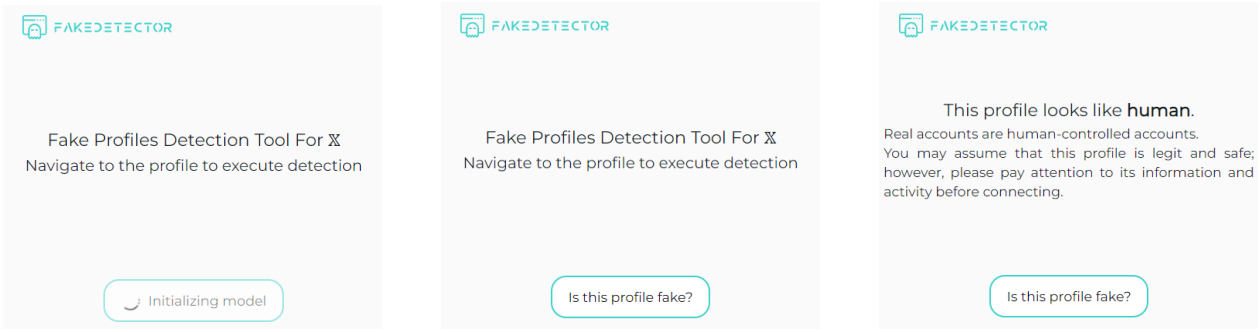


Figure 10. System implementing fake account detection approach.

To summarize the components of the tool outlined so far, we could present the process of its usage in the following steps:

1. Upon first launching the extension, the user navigates to the X social network profile of their interest (Figure 11a);
2. Within the extension, the user selects the button to detect if the account is fake (Figure 11b);
3. The extension takes a screenshot of the web page element containing the profile on the portal;
4. The image is sent to the facade component by the WebSocket API;
5. The facade forwards the image as the input to the machine learning model;
6. The model predicts whether the analyzed account is fake and returns the results;
7. Through the facade, the results are sent back to the end user (Figure 11c);
8. The probability of the account being fake is displayed to the user in the extension interface.

Figure 11 depicts the extension's interfaces in its particular states, such as ML model initialization (Figure 11a), its readiness when the ML model is already loaded (Figure 11b), and the detection result produced by the ML model (Figure 11c).



(**a**) ML model initialization process (**b**) ML model once being loaded (**c**) ML model detection output

Figure 11. Extension's interfaces in its particular states.

4.3. Tests Performed on Fake Account Detection Tool

This subsection provides the results of the tests conducted on a tool for detecting fake profiles in a social network for both the implemented copy and the original environment—the X portal. The predictions obtained from the machine learning model were placed into the confusion matrix; then, the true positive rate was calculated.

4.3.1. Testing in Implemented Copy of X Environment

The implemented environment is a copy of X's web view created for dataset generation purposes needed to train the ML classifier. The features and characteristics of the profiles were appropriately adjusted to the definition of their class. Therefore, since it most closely resembles the training dataset, high efficiency rates were expected there.

A new test set containing unique images of user profiles (1000 per class) was generated to verify the accuracy of the classification tool. The resulting confusion matrix is shown in Table 5.

Table 5. Confusion matrix of the proposed classification tested on the implemented copy of the X environment.

		Classified				**Total**	**True Pos. %**
		Bot	**Cyborg**	**Human**	**Verified**		
Actual	Bot	992	0	8	0	1000	99.2%
	Cyborg	24	956	20	0	1000	95.6%
	Human	2	1	997	0	1000	99.7%
	Verified	0	18	58	924	1000	92.4%

4.3.2. Testing in Original X Environment

Tests of the fake profile detection tool on the original X platform were conducted manually from the end user perspective. For this reason, the test set was significantly reduced. It is also worth noting that the initial classification of accounts could not be considered objective in the case of the bots, cyborgs, and real users classes, as they were selected on a profile appearance basis. Therefore, the result of the presented tests is only illustrative, proving the satisfactory functioning of the tool and detecting the identified reasons for incorrect model classifications.

To verify the classification tool's performance, unique X user accounts were manually selected (50 per class). Moreover, as the model was trained on a set of accounts that do not have posts with videos or photos, we preferred to choose accounts with recent text-based posts when conducting the test. The resulting confusion matrix is shown in Table 6.

Table 6. Confusion matrix of the proposed classification tested on the original X environment.

		Classified				Total	True Pos. %
		Bot	**Cyborg**	**Human**	**Verified**		
Actual	Bot	46	0	4	0	50	92%
	Cyborg	0	45	5	0	50	90%
	Human	0	13	37	0	50	74%
	Verified	0	6	12	32	50	64%

5. Discussion

With fake accounts exposing social networks to considerable danger and frequently being misused by attackers, more detection solutions are being developed (e.g., LinkedIn, Facebook, and X). The statistics of these tools provided by the creators of the OSN allow us to estimate their effectiveness; however, we still need to learn the details of their implementation. On the other hand, it is known that the vast majority use machine learning models for detection.

Our study significantly differs from most approaches to detecting fake accounts, as it analyzes the outlook of a user's account, thus focusing on its most essential features. Nevertheless, there is room for improvement in this tool's performance and efficiency. After analyzing the obtained test results, we noted slight misclassification in the bot class with the human class due to the overlapping percentage of accounts characterized by minimal personal information, confirmed during manual tests. The informal definition of the cyborg class led to 2.4% of profiles being mistakenly classified as bots and 2% as humans, since "cyborg" is a human-assisted "bot". The classification model performed quite differently in the case of the original X environment, showing us 10% of cyborgs being classified as humans, since the activity level of some "cyborgs" may have been similar to its extraordinarily high level of real users. The same situation also arises for the human class, with the 74% of correctly detected data, even though the classification tested on the implemented environment ended up with the best true positive rate of 99.7%. The class of verified profiles both on the copy and the original environments was the most misunderstood by the trained model. First of all, the reason for this behavior of the model is the confusing activity level of humans and cyborgs. To make class detection work as initially intended based on the presence of a verification icon, it would be necessary to increase the weight of this factor.

Following the statistics mentioned above, the main reason for the mistakes is that the class definitions did not precisely correspond to the profile activity taking place on X. Hence, it would be essential to create more precise features of the image database to obtain the most accurate match to the current state of the social network and expand user profile views to include posts with graphical content (images and videos). However, this was not the study's primary objective, as we mainly intended to introduce the applicability of image classification to the fake account detection problem by demonstrating the effectiveness of this approach.

This study is limited in a certain way, as it uses predefined account types to generate the image dataset, which may need to be revised in light of the social network's real behavior and appearance. First and foremost, this is indicated by the statistics obtained during the test analysis of the detection tool, where the model performed much better in the environment according to which it was taught. Online social networks are being actively developed, and a sudden change in any of the elements of the environment under study can have a negative impact on the accuracy of the detection method.

A further challenge may be the high sensibility of the created plugin tool to possible browser or system over-scaling, as well as the incorrect position of the profile in the browser window, causing the information of the account to be shifted. Given the entire spectrum of existing devices, modes, and fonts, supporting and correctly addressing every modification

would require much effort. Although we can extend the tool to support different types of devices or include other modes and fonts, the most reliable solution would be to require users to employ the specified setup. Any exceptions to supported settings may be followed up with a message to the user about existing limitations and recommended setup to ensure detection effectiveness.

6. Conclusions

In this article, we presented the results of our work on detecting fake accounts on the X platform. In this case study, we focused on defining and classifying features of fake profiles to create a dataset with 15,000 images of unique accounts. This provides an opportunity to pave the way for further research.

Moreover, we conducted a comparative analysis of classification algorithms, selecting the most suitable one. An extension for the Google Chrome browser was created to use the trained model, allowing for the real-time interactive analysis of accounts. Finally, the initial tests were conducted, which concluded that the classes of bot and cyborg accounts were found to have the highest classification accuracy. Profiles belonging to groups of real and verified users were relatively often confused with each other, which is acceptable from the perspective of the tool's primary purpose. The unpredictable behavior of the detection model is due to the limitations encountered.

In future work, we plan to expand research on image-based account detection to other social networks (where possible). Moreover, we plan to verify whether using different types of neural networks will increase the detection level of fake accounts and their kind. Finally, due to the dynamic nature of social network web pages, the solution should be continuously adapted to the most up-to-date version of the network.

Regarding the stability of the created solution, it is worthwhile to reduce the sensitivity of the browser plugin, so the screenshot would not be taken according to fixed parameters but against a specific page element identifier that contains the necessary data for correct analysis. Moreover, despite X's default settings, it is worth supporting the most basic settings (e.g., system fonts and light and dark modes) and including support for at least two device types—desktop and mobile.

Author Contributions: Conceptualization, W.D. and M.S.; methodology, W.D.; software, W.D.; validation, W.D. and M.S.; formal analysis, W.D.; writing—original draft preparation, W.D. and M.S.; visualization, W.D. and M.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Meltwater, W.A.S. Digital 2023 Global Overview Report. Available online: https://datareportal.com/reports/digital-2023-global-overview-report (accessed on 13 June 2024).
2. Almadhoor, L. Social media and cybercrimes. *Turk. J. Comput. Math. Educ. (TURCOMAT)* **2021**, *12*, 2972–2981.
3. Di Domenico, G.; Sit, J.; Ishizaka, A.; Nunan, D. Fake news, social media and marketing: A systematic review. *J. Bus. Res.* **2021**, *124*, 329–334. [CrossRef]
4. Shu, K.; Bhattacharjee, A.; Alatawi, F.; Nazer, T.H.; Ding, K.; Karami, M.; Liu, H. Combating disinformation in a social media age. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2020**, *10*, 1–39. [CrossRef]
5. Social Media Use Statistics. Available online: https://gs.statcounter.com/social-media-stats (accessed on 4 March 2024).
6. Umbrani, K.; Shah, D.; Pile, A.; Jain, A. Fake Profile Detection Using Machine Learning. In Proceedings of the 2024 ASU International Conference in Emerging Technologies for Sustainability and Intelligent Systems (ICETSIS), Manama, Bahrain, 28–29 January 2024; pp. 966–973. [CrossRef]
7. Durga, P.; Sudhakar, D.T. The use of supervised machine learning classifiers for the detection of fake Instagram accounts. *J. Pharm. Negat. Results* **2023**, *14*, 267–279. [CrossRef]

8. Prakash, O.; Kumar, R. Fake Account Detection in Social Networks with Supervised Machine Learning. In *International Conference on IoT, Intelligent Computing and Security. Lecture Notes in Electrical Engineering*; Agrawal, R., Mitra, P., Pal, A., Sharma Gaur, M., Eds.; Springer: Singapore, 2023; Volume 982, pp. 287–295. [CrossRef]
9. Kanagavalli, N.; Sankaralingam, B.P. Social Networks Fake Account and Fake News Identification with Reliable Deep Learning. *Intell. Autom. Soft Comput.* **2022**, *33*, 191–205. [CrossRef]
10. Bhattacharyya, A.; Kulkarni, A. Machine Learning-Based Detection and Categorization of Malicious Accounts on Social Media. In *Social Computing and Social Media. HCII 2024. Lecture Notes in Computer Science*; Coman, A., Vasilache, S., Eds.; Springer: Cham, Switzerland, 2024; Volume 14703, pp. 328–337. [CrossRef]
11. Goyal, B.; Gill, N.S.; Gulia, P.; Prakash, O.; Priyadarshini, I.; Sharma, R.; Obaid, A.J.; Yadav, K. Detection of Fake Accounts on Social Media Using Multimodal Data With Deep Learning. *IEEE Trans. Comput. Soc. Syst.* **2023**, 1–12. [CrossRef]
12. Wani, M.A.; Agarwal, N.; Jabin, S.; Hussain, S.Z. Analyzing real and fake users in Facebook network based on emotions. In Proceedings of the 2019 11th International Conference on Communication Systems & Networks (COMSNETS), Bengaluru, India, 7–11 January 2019; pp. 110–117. [CrossRef]
13. Gupta, A.; Kaushal, R. Towards detecting fake user accounts in facebook. In Proceedings of the 2017 ISEA Asia Security and Privacy (ISEASP), Surat, India, 29 January–1 February 2017; pp. 1–6. [CrossRef]
14. Boshmaf, Y.; Logothetis, D.; Siganos, G.; Lería, J.; Lorenzo, J.; Ripeanu, M.; Beznosov, K. Integro: Leveraging victim prediction for robust fake account detection in OSNs. In Proceedings of the Network and Distributed System Security Symposium 2015 (NDSS'15), San Diego, CA, USA, 8–11 February 2015; pp. 1–15. [CrossRef]
15. Conti, M.; Poovendran, R.; Secchiero, M. Fakebook: Detecting fake profiles in on-line social networks. In Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining, Istanbul, Turkey, 26–29 August 2012; pp. 1071–1078. [CrossRef]
16. Sheikhi, S. An Efficient Method for Detection of Fake Accounts on the Instagram Platform. *Rev. d'Intell. Artif.* **2020**, *34*, 429–436. [CrossRef]
17. Akyon, F.C.; Esat Kalfaoglu, M. Instagram Fake and Automated Account Detection. In Proceedings of the 2019 Innovations in Intelligent Systems and Applications Conference (ASYU), Izmir, Turkey, 31 October–2 November 2019; pp. 1–7. [CrossRef]
18. Zarei, K.; Farahbakhsh, R.; Crespi, N. Deep dive on politician impersonating accounts in social media. In Proceedings of the 2019 IEEE Symposium on Computers and Communications (ISCC), Barcelona, Spain, 29 June–3 July 2019; pp. 1–6. [CrossRef]
19. Harris, P.; Gojal, J.; Chitra, R.; Anithra, S. Fake Instagram Profile Identification and Classification using Machine Learning. In Proceedings of the 2021 2nd Global Conference for Advancement in Technology (GCAT), Bangalore, India, 1–3 October 2021; pp. 1–5. [CrossRef]
20. Adikari, S.; Dutta, K. Identifying fake profiles in linkedin. In Proceedings of the Pacific Asia Conference on Information Systems (PACIS), Chengdu, China, 24–28 June 2014; pp. 1–30. [CrossRef]
21. Xiao, C.; Freeman, D.; Hwa, T. Detecting Clusters of Fake Accounts in Online Social Networks. In Proceedings of the 8th ACM Workshop on Artificial Intelligence and Security, Denver, CO, USA, 16 October 2015; pp. 91–101. [CrossRef]
22. Rostami, R.R. Detecting Fake Accounts on Twitter Social Network Using Multi-Objective Hybrid Feature Selection Approach. *Webology* **2020**, *17*, 1–18. [CrossRef]
23. Sahoo, S.R.; Gupta, B.B. Real-Time Detection of Fake Account in Twitter Using Machine-Learning Approach. In *Advances in Computational Intelligence and Communication Technology. Advances in Intelligent Systems and Computing*; Springer: Singapore, 2020; Volume 1086, pp. 149–159. [CrossRef]
24. Prabhu Kavin, B.; Karki, S.; Hemalatha, S.; Singh, D.; Vijayalakshmi, R.; Thangamani, M.; Haleem, S.L.A.; Jose, D.; Tirth, V.; Kshirsagar, P.R.; et al. Machine Learning-Based Secure Data Acquisition for Fake Accounts Detection in Future Mobile Communication Networks. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 6356152. [CrossRef]
25. Van Der Walt, E.; Eloff, J. Using Machine Learning to Detect Fake Identities: Bots vs Humans. *IEEE Access* **2018**, *6*, 6540–6549. [CrossRef]
26. Roy, P.K.; Chahar, S. Fake Profile Detection on Social Networking Websites: A Comprehensive Review. *IEEE Trans. Artif. Intell.* **2020**, *1*, 271–285. [CrossRef]
27. Krishnamurthy, B.; Gill, P.; Arlitt, M.F. A few chirps about twitter. In Proceedings of the WOSN '08: Proceedings of the First Workshop on Online Social Networks, Seattle, WA, USA, 17–22 August 2008; pp. 19–24. [CrossRef]
28. Chu, Z.; Gianvecchio, S.; Wang, H.; Jajodia, S. Who is Tweeting on Twitter: Human, Bot, or Cyborg? In Proceedings of the 26th Annual Computer Security Applications Conference, Austin, TX, USA, 6–10 December 2010; pp. 21–30. [CrossRef]
29. Meta Open Source React. The Library for Web and Native User Interfaces. 2023. Available online: https://react.dev/ (accessed on 13 June 2024).
30. Microsoft Corp. Playwright. 2023. Available online: https://playwright.dev/ (accessed on 13 June 2024).
31. Faker Open Source, Faker. 2023. Available online: https://fakerjs.dev/ (accessed on 13 June 2024).
32. Marby, D.; Yonskai, N. Lorem Picsum. Images. 2023. Available online: https://picsum.photos/images (accessed on 13 June 2024).
33. Khaled, S.; El-Tazi, N.; Mokhtar, H.M.O. Detecting Fake Accounts on Social Media. In Proceedings of the IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 10–13 December 2018; pp. 3672–3681. [CrossRef]
34. O'shea, K.; Nash, R. An introduction to convolutional neural networks. *arXiv* **2015**, arXiv:1511.08458. [CrossRef]
35. Pal, M. Random forest classifier for remote sensing classification. *Int. J. Remote. Sens.* **2005**, *26*, 217–222. [CrossRef]

36. Rish, I. An empirical study of the naive Bayes classifier. In Proceedings of the IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence, Seattle, WA, USA, 4–6 August 2001; Volume 3, pp. 41–46.
37. Lubbers, P.; Albers, B.; Salim, F. Using the WebSocket API. In *Pro HTML5 Programming*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 159–191. [CrossRef]
38. Interfejs API Chrome.tabs, On-Line Documentation. Available online: https://developer.chrome.com/docs/extensions/reference/api/tabs (accessed on 4 March 2024).