future internet

MDPI

Article

Predicting Rogue Content and Arabic Spammers on Twitter

Adel R. Alharbi [1,*] and Amer Aljaedi [2,*]

[1] Department of Computer Engineering, University of Tabuk, Tabuk 71491, Saudi Arabia
[2] Department of Information Technology, University of Tabuk, Tabuk 71491, Saudi Arabia
* Correspondence: aalharbi@ut.edu.sa (A.R.A.); aaljaedi@ut.edu.sa (A.A.)

Received: 4 October 2019; Accepted: 28 October 2019; Published: 30 October 2019

Abstract: Twitter is one of the most popular online social networks for spreading propaganda and words in the Arab region. Spammers are now creating rogue accounts to distribute adult content through Arabic tweets that Arabic norms and cultures prohibit. Arab governments are facing a huge challenge in the detection of these accounts. Researchers have extensively studied English spam on online social networks, while to date, social network spam in other languages has been completely ignored. In our previous study, we estimated that rogue and spam content accounted for approximately three quarters of all content with Arabic trending hashtags in Saudi Arabia. This alarming rate, supported by autonomous concurrent estimates, highlights the urgent need to develop adaptive spam detection methods. In this work, we collected a pure data set from spam accounts producing Arabic tweets. We applied lightweight feature engineering based on rogue content and user profiles. The 47 generated features were analyzed, and the best features were selected. Our performance results show that the random forest classification algorithm with 16 features performs best, with accuracy rates greater than 90%.

Keywords: arabic text classification; online social network; twitter; machine Learning; spam detection; rogue contents

1. Introduction

Twitter is a platform that allows users to compose messages of 140 characters or less. These messages are known as tweets and can include text, short videos, images, and hyperlinks. Twitter usernames start with the prefix @. Users of Twitter build their social networks by interacting with fans and followers. Tweets generated by users appear on their homepage and the timelines of their followers, and they can be discovered by Twitter's search engine. The tweets are often spread by followers who click the "Retweet" icon. The tweet is often relayed continuously, as is the username prefixed by @, which is included in the tweet. The subjects of tweets are frequently indexed by hashtags related to each subject. All Twitter hashtags are preceded by the hash (#) symbol and can even be found by using Twitter's search engine [1].

Twitter is a popular online social network (OSN) in Middle Eastern countries, especially Gulf countries such as Saudi Arabia, which ranks second for the number of Twitter users worldwide. In our previous investigation, we found that about three quarters of the tweets with trending hashtags in Saudi Arabia were spam messages. This estimation is backed by independent reports that place Saudi Arabia as the second most common "global target for online spam and other forms of cyber-violation" and as the "most spammed country in the world for three years in a row with an 83.3% spam rate" [2]. A deeper analysis showed an even higher percentage of automatically generated tweets. Therefore, in addition to Twitter's resources being consumed by malicious accounts rather than intended users, any statistics or opinions mined using these trends are unreliable. In particular, it is logical to question

any reports that indicate Saudi Arabia as the Arab nation with the highest number of active Twitter users [3] or that show "booming usage" with a penetration higher than 51% [4].

Much research has been conducted on data mining and machine learning for the English corpus [5–7], while little research has been carried out on Arabic text. This is mainly because of its morphological complexity and the limited availability of compatible Arabic language software. In addition, Arabic words have different meanings and spellings, the sentence structure of Arabic differs from that of English, Arabic letters have different shapes depending on the letter location in a word, and Arabic words are either masculine or feminine and come in three different formats: singular, dual, or plural [8].

Arabic nations are facing challenges in the detection of rogue accounts and spam in Arabic tweets. The use of keyword lists to identify spam accounts approach used by current censorship systems implemented in Arab nations [9]. In this paper, we focus on classifying rogue and spam content in Arabic tweets using machine learning algorithms. We explain the data collection approach and the features engineered. Then, we present the evaluation of our proposed Arabic approach with a random forest classifier, and we report the minimum number of features that can produce the best results. Much research has been conducted on the detection of spammers in the English language, and comparatively little attention has been paid to determining the minimum number of features that give approximately the same result.

We summarize the contributions of this work in the following points:

- We obtained one of the largest data sets for rogue and spam accounts with Arabic tweets by directly collecting the tweets from such accounts.
- We surveyed more than 180 Arabian participants by administering Twitter security-related questionnaires to measure and discover the area of security that could be of the most concern to Twitter users.
- We enhanced and engineered 47 of the most effective and simple features based on tweets, profile content, and social graphs.
- Using the random forest feature selection method, we evaluated and compared different numbers of features and found that the method with 16 features performed best.
- Using the random forest classifier, we evaluated and compared different numbers of variables, which were randomly sampled as candidates at each split, and we found that the method with 8 variables performed best.
- Using the random forest classifier, we evaluated and compared different numbers of trees and found that 2500 trees performed best.

The rest of the paper is organized as follows: Section 2 discusses the motivation behind our research. Section 3 reviews the most significant studies on this subject. Next, Section 4 explains our proposed system implementation, and Section 5 describes the collection and labeling of tweet data. Then, Section 6 explains the steps of pre-filtering the collected tweet data. Section 7 describes the extracted features used for spam detection purposes from each Twitter user account. Section 8 analyzes the generated features by ranking them and examines their relationships. Section 9 describes the supervised classification algorithm, and Section 10 shows the experimental results of our proposed approach using several classification metrics. Finally, Section 11 concludes the paper and suggests ideas for future work.

2. Motivation

This section discusses the motivation behind our research work, with real examples of tweets with Arabic rogue and spam content and a public survey of Twitter users.

Twitter has defined spamming behavior and the rules of tweet freedom [2,10,11]. We established our definition of a spam tweet. A tweet is considered a spam tweet if it satisfies the following conditions:

- The tweet is advertising a fake or unreal product or service;
- The tweet contains rogue and spam verbs or words in the form of text, images, or videos [10];

- The URL redirects to a malicious or phishing website [2];
- The URL redirects to a page that is not tweet-related.

Figure 1 shows three examples of spam tweets that are sometimes found with unrelated hashtags. Spammers sell fake Netflix accounts, as in the first tweet: 'We sell Netflix accounts [phone number] on the occasion of opening the store. Two months for the price of one month for 20 riyals. Offer expires 24 h from now.' Furthermore, some spammers claim to sell educational certificates from accredited universities (such as bachelor's, master's, doctorate, and IELTSL degrees) without the buyer attending the institution or undergoing any tests. The middle tweet says 'Accredited university degrees for sale; Bachelor's degrees for sale in Saudi Arabia; IELTSL accredited certificates for sale; Doctorate degree for sale; Master's distance certificates for sale.' In the last example in the figure, the goal is to sell followers to social media users, along with a local phone number for interested customers: 'Selling followers on Twitter, Instagram, YouTube, and all social media services. Call [phone number].'

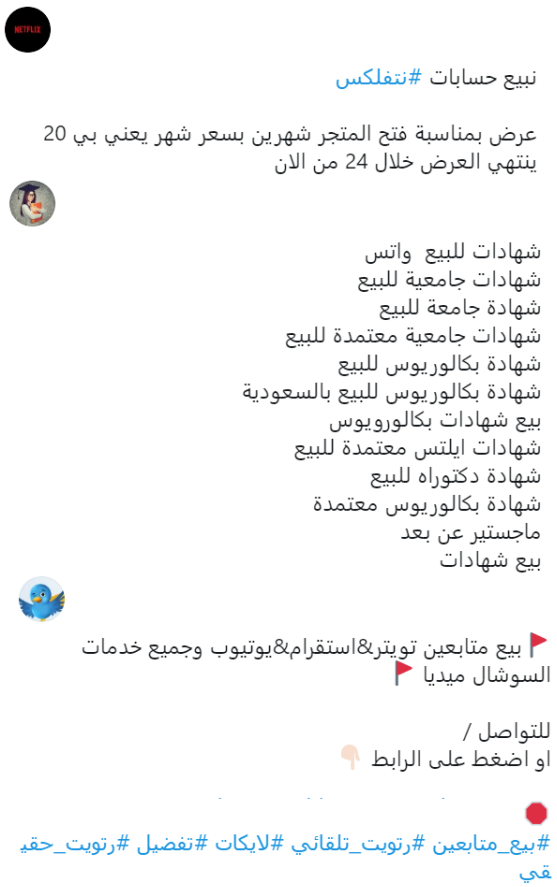

Figure 1. Three examples of spam tweets that sell Netflix accounts, educational certificates, and social media followers.

Some of the data for this project were gathered by distributing a questionnaire to a small sample of a local community in Saudi Arabia. The main objective of the survey was to measure the risks to information security when using Twitter. Hence, we surveyed around 189 respondents, 91% of whom were born in Saudi Arabia. The rest of the respondents were from different countries, namely the United States (7%), Britain (2%), Iran, Belgium, and Egypt (1%). Figure 2 presents multiple pie charts, with each chart showing the percentage of each answer to one question in the survey.

The participants were 85% male and 15% female. Most of the participants had a bachelor's degree (51%), followed by those with a master's degree (25%), a doctorate degree (13%), and a high school degree (11%). Most of the participants were 21–30 years old (29%), followed by 11–20 years old

(27%), 31–40 years old (25%), and 41–50 years old (19%). In response to the number of years that the participants had been using Twitter, 35% of them had been using the platform for 4–6 years, followed by 25% for 7–9 years, 20% for 1–3 years, 16% for less than 1 year, and 4% for 10–12 years. On the subject of the number of hours of Twitter use per day, 46% responded that they use it for less than an hour, 41% of them responded that they use it for 1–3 h, 11% of them responded that they use it for 4–6 h, and 2% of them responded that they use it for 7–9 h.

Participants' opinions about security-related questions were also collected. They were asked to assess the level of Twitter security as measured by tweets that contain no viruses or fake links. Of the responses, 35% graded the security level as medium, 19% high, 17% very low and low, and 11% very high. They also assessed the level of credibility of the information posted on Twitter, with 48% reporting medium, followed by 21% low, 16% very low, 11% high, and 5% very high. They voted on the presence of immoral and inappropriate videos, with 28% very high, 23% high, 22% medium, 15% very low, and 12% low. They estimated the percentage of fraudulent accounts on Twitter, and 33% indicated very high, 22% high, 22% medium, 10% low, and 9% very low.

Most of the participants (94%) answered that they had never been cheated, financially defrauded, or blackmailed on Twitter, while the remaining 6% stated otherwise. Only 8% of them responded that their Twitter account had been hacked, whereas 92% indicated that they had not been hacked. The majority (88%) of the respondents had unverified Twitter accounts (i.e., the account has a blue checkmark), whereas only 12% were verified.

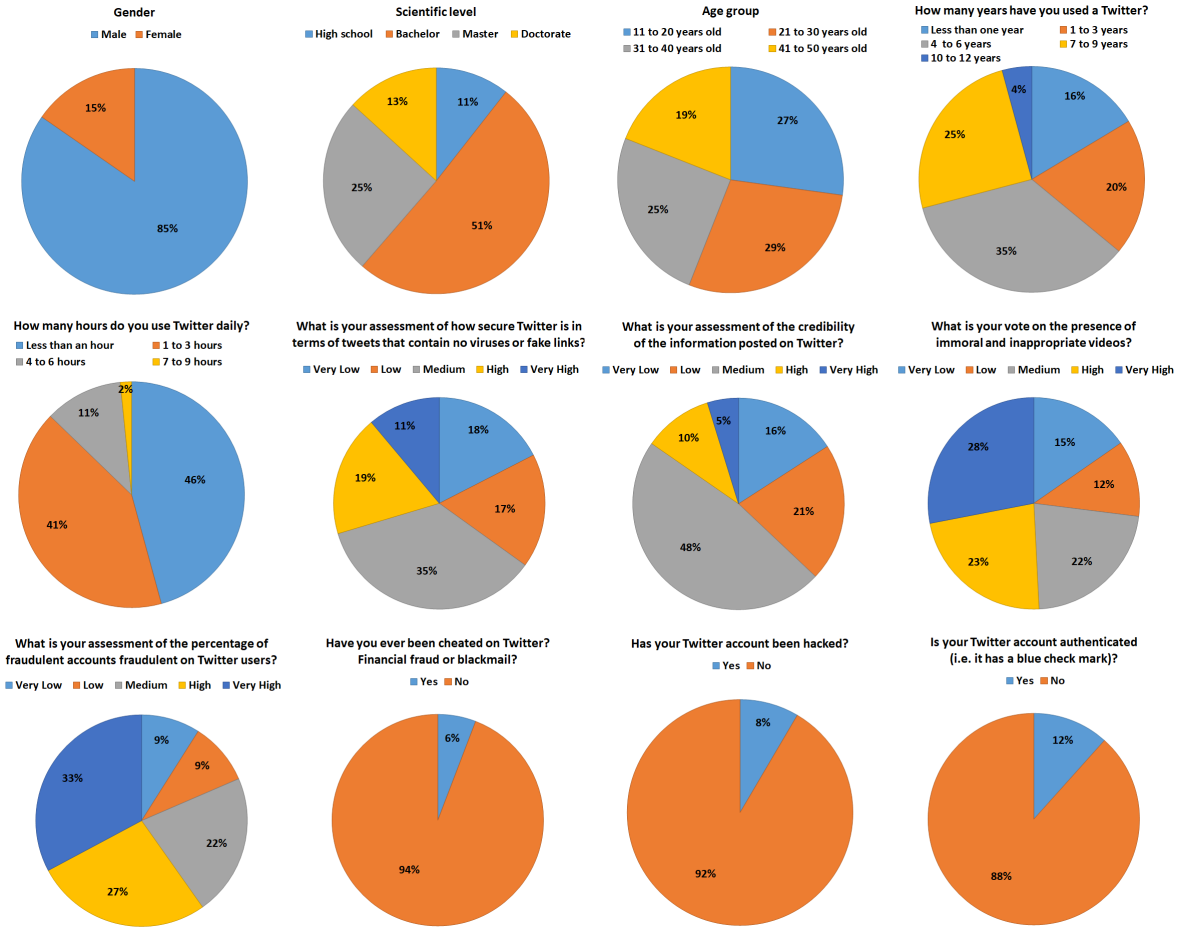

Figure 2. Participants' responses to the questionnaire items.

3. Literature Review

This section reviews the most significant studies on spam detection and machine learning related to users of Twitter.

One of the most famous hallmarks of the digital age is the popularization of social media sites. Individuals use these platforms to communicate with family, friends, and even business associates. Digital social platforms suffer from spamming because users with unethical intent use them for self-serving reasons. Commonly targeted sites include "Twitter, Facebook, and LinkedIn" [12]. The main types of spam include the duplication of information, continuous reposts of topics for marketing purposes, and distribution of unwanted messages. The present literature review discusses how machine learning can be used to recognize spam from Twitter accounts that are predominantly in Arabic languages. All the algorithms and associated approaches strive to detect whether the posted text has positive or negative sentiments, together with the extent of profanity [13]. There are other reasons to examine social media posts, especially in the modern age, in which everything is linked to cyberspace. The aim of this review is to focus on Arabic Twitter spam and the use of machine learning as a possible solution.

Twitter is among the most frequently used sites, and it continues to grow with each passing year. Its large number of users also implies a high percentage of spam messages [6]. Twitter allows short messages with only text and HTTP links available to users. Individuals who spread spam content rely on their anonymity to communicate with legitimate users. It has become crucial for developers to establish ways to prevent spam content. Among the many forms of doing so is the use of traditional classifiers for detecting spam; these include random forest (RF), naïve Bayesian (NB), support vector machine (SVM), and *k*-nearest neighbor (*K*-NN) schemes [6]. The two researchers found that random forest was the best method for this purpose, with a precision of 95.7% on a small Twitter data set. The process is driven by the growing need for spam detection by information filtering. Under the Twitter umbrella, researchers have noted that the limited length of posts does not provide the required number of word occurrences. Thus, traditional methods, such as "bag-of-words", are not ideal [14]. This discovery marked the beginning of more sophisticated techniques.

On a structural level, spam detection entails checking for certain words in a tweet. The same concept applies regardless of whether a person decides to use simple or complex machine learning systems. Mubarak et al. [15] provided a simple means of understanding the concept. People may choose to filter information for various reasons, such as the need to categorize information, remove pornographic content from the media stream, or prevent children from seeing specific posted messages. All these objectives lead to machine learning interactions with the Twitter API and other interfaces. A more in-depth analysis of spamming problems reveals engineering algorithms such as NB IBK (which is may refer to Ibk algorithm, implements the *k*-NN algorithm) as means of finding solutions to the problem. As observed by McCord and Chuah [6], a person can pick the best algorithm to use though empirical examination. Interestingly, Ameen and Kaya [12] carried out a similar exercise and found that random forest had the highest success at 92.95%. A researcher must experiment to determine the best algorithm to use before proceeding with further analysis. There is no particular algorithm that surpasses all others under all circumstances; this explains the need to experiment with various approaches.

Before moving to advanced classifier techniques, it is essential to understand the reason that most researchers have dismissed SVM classifiers such as bag-of-words and bag-of-means. Alshehri et al. [16] used hashtags and N-grams to screen out adult Arabic content. The bag-of-words method uses binary values to check for certain words in a posted text, while bag-of-means involves finding an average of word vectors. The result of their research was a 79% accuracy of processing. This precision is extremely low compared with the results of the random forest approach proposed by McCord and Chuah [6]. The precision of their RF method was 95.7%. McCord and Chuah also used SVM but determined that it was comparatively ineffective. As illustrated above, these simple detection methods are ineffective because Twitter allows for only a short messages: tweets have a limit of 140 characters, which means

that the probability of finding a mean of word vectors is low. Nevertheless, SVM was able to attach abusive tweets to geographical targets. This finding laid a foundation for future research involving better-performing machine learning strategies.

Understanding the shift from simple detection techniques to modern advanced designs is central in spam detection. First, Sriram and his colleagues developed a simple scheme to classify digital content using several categories: "News (N), Events (E), Opinions (0), Deals (D), and Private Messages (PM)" [14]. The method typically uses the user information and the details found in the messages. It is a technique that filters spam and allows Twitter users to see only the tweets associated with their background. It uses a similarity search approach, which can benefit users of social media sites. However, the method is prone to noise and can be ineffective in certain situations. The present assessment focuses on filtering Arabic content on Twitter, whereas most research on this topic has focused on the English language, which uses the Roman alphabet. Al-Eidan et al. [17] provided two methods with the aim of evaluating Arabic content; the first of these proposed systems applies "the similarity between Twitter posts and authentic news sources, while the second approach uses a set of proposed features" [17]. These techniques sound similar to the approach proposed by Sriram et al. [14] with classes. However, the design offered by Al-Eidan covers more categories, including Credible, Not Credible, and Questionable, with the use of light stemming. The systematic approach aids in developing an application to be included in a browser setting or the Twitter API.

After this, the basic knowledge of mechanisms by which machine learning interacts with spam detection objectives is explained. The next area of interest is the benefit of using machine learning to stop abusive accounts, primarily those from users with an Arabic association. As noted earlier, filtering techniques in Twitter rely on the Roman alphabetic system and are useless for Arabic tweets. McCord and Chuah [6] proposed RF because of its outstanding merits, while the other authors cited above aimed to establish a kind of similarity method that uses classes. Abozinadah et al. [1] described the Arabic language as a complex system that can be broken down into symbols and letters with gender undertones and various formats and spellings. As a result, few research studies exist that offer information filtering for digital platforms for this language. They used a similar classification to that of McCord and Chuah but added a data set that classified information into "accounts, tweets, hashes, links (URLs)" [1]. The system relies on *k*-NN, NB, and decision tree (DT) to detect controversial language terms, such as cursing and pornography. Their search found that NB was the most preferable, and its accuracy was 90%. Abozinadah and his colleagues realized that reviewing both the user profile and the posted content was an excellent foundation for spam detection. They also proposed translating Arabic words into English for ease of evaluation.

Modern research supports advanced spam detection approaches based on accurate measurements. Among the above studies, one found that RF was ideal, with an accuracy of 95.7% [6], while another favored NB and had a 90% accuracy [1]. Additionally, research by Rsheed and Khan [18] compared DT, NB, and rule-based W-JRIP (which is a type of propositional rule learning and repeated incremental pruning to produce error reduction (RIPPER) proposed by Willeam W), and they chose DT as the most suitable classifier. Their preferred choice achieved an accuracy of 85.6% when coupled with light stemming tactics. Their findings proved that different research approaches yield different results; however, advanced machine learning classifiers provided better accuracy than the simple classifiers. This study showed that the proposed techniques have functional versatility. They used these classifier methods to check the popularity of news articles on Arabic Twitter. For example, apart from the statistical figures found, they realized that users reviewed the website and the source of the material to rate their interest in the article. Arabic users checked the article source more than they checked the title and description of the content. This implies that they value honest news more than the vocabulary used.

Another study that focused on Arabic spam accounts helped decipher the reason that some of these accounts go undetected. El-Mawass and Alaboodi [2] provided a realistic view of spamming in Saudi Arabia and some tactics used by spammers to avoid detection. Consequently, to find these

abusive accounts, they developed an approach to handling these methods of evasion. These two authors found that Saudi Arabia had about 75% of the spam content found worldwide [2]. It has become imperative to find ways to reduce this figure through customized maneuvering. Furthermore, the country suffers from economic and political challenges as a result of harmful spam in social media. Evasion techniques include "using semantic synonyms to texts, buying followers, the use of clusters formed by spammers, and mimicking the behavior of authentic accounts" [2]. These strategies lower the accuracy of currently used machine learning techniques. This discovery means that the next generation of classifiers needs to include these evasion techniques as part of the accuracy measurement.

The above literature review uses numerous credible sources to explain the concept of spam detection and its use in combating profanity in Arabic Twitter. There have been efforts to include other purposes of information filtering, such as marketing purposes and child protection. However, the main goal has remained to establish a machine learning approach that can detect certain words and trends. The literature review mainly focuses on the Arabic language because of the limited number of studies carried out to monitor Arabic Twitter. Saudi Arabia generates three quarters of the Twitter spam found in the world. The analysis also reveals that advanced machine learning algorithms, such as RF, DT, and NB, are more effective compared with simple algorithms such as bag-of-words. Lastly, the Arabic language provides a challenge for web developers because of its complexity in spelling, symbolism, and the existence of different dialects across the world.

As a comparison between the related work approaches that attempted to detect rogue contents and Arabic spammers on Twitter with our proposed approach based on the results and novelty keys. To the best of our knowledge, we found about three similar related studies to ours. The state of the artwork by Abozinadah et al. [1] gathered 1,300,000 tweets and they extracted more than 1400 number of features, where they found that 100 features have a better performance than a larger number of features. They compared three classification models: NB, SVM, and DT, where the NB classifier had the best performance with 10 tweets and 100 features, which achieved with 90% accuracy rate. Another important work was by El-Mawass and Alaboodi [2] where they used a large collected data set of over than 23 million Arabic tweets, and a manually labeled sample of over than 5000 tweets. They randomly chose 10% of the 55,239 obtained tweets for classification and validation. They generated about 20 features, then they selected the top 14 features based on applying the information gain and the Chi squared selection techniques. They compared three classification algorithms: NB, RF, and SVM. They achieved with 92.59% accuracy rate by using the RF classifier. Finally, the work by Alshehri et al. [16] used the same technique of El-Mawass and Alaboodi work [2] to collect their data set by collecting a list of hashtags discusses rogue contents to obtained tweets. The data size was about 27 million tweets and they ended up with a total of 200,000 tweets after the pre-processing step. They developed two models: bag-of-words (BOW) and bag-of-means (BOM). They achieved with 79% accuracy result by using BOM model.

However, all the similar related studies were limited to the number of the collected tweets. Most of the studies extracted their features based on Tokenization technique for the collected tweet texts. In documents written in English and French, the Tokenization process can be used as these languages use white space to differentiate the words. For example, Chinese, Thai, Hindi, Urdu, Tamil, etc., the Tokenization method could not be used in other languages [19]. The work by Alshehri et al. [16] indicated that there is another method to extract the features based on such as: bag-of-words (BOW) and bag-of-means models which are suitable for language texts as Arabic. In addition, the work by El-Mawass and Alaboodi [2] pointed that RF classifier as a model could be a great choose because it performed better than other classifiers as mentioned earlier.

Nevertheless, our work collected a large-scale data set of rogue contents and Arabic spammers on Twitter close to 3 million tweets. The extracted features were completely statistic-based and used the bag-of-words method for the tweet texts. The features are simple and could be easily programmed in the Twitter platform itself. The proposed approach used RF classifier as both feature selection and

classification model techniques, therefore there is less computation performance. In top of that the accuracy rate was 93.21% for the selected top 16 features.

4. Proposed Solution

Our suggested scheme can be introduced as an independent implementation that depends on machine learning (ML) techniques to detect rogue contents and Arabic spammer tweets. Figure 3 illustrates the proposed system architecture. The general idea that underlies our system is to use Twitter as a public platform to capture users' tweet data by using the Twitter API [20]. Data are collected in a database and then directly forwarded to implement a data pre-processing step that contains several pre-filter functions, which are needed to manage the variable and unpredictable nature of the language used in tweets. Most tweets are extremely likely to contain some type of grammar or spelling errors, emoticons, symbols, pictographs, and flags.

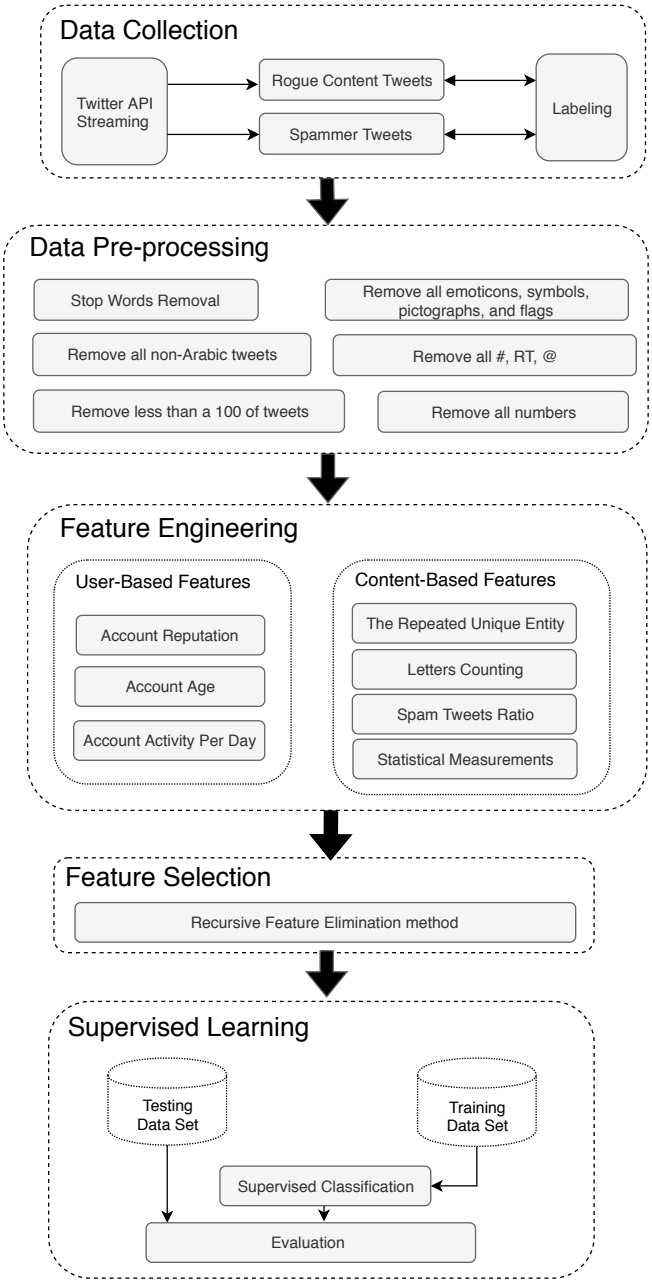

Figure 3. The architecture of the proposed system.

After pre-filtering, the system applies our developed lightweight feature engineering to extract useful information. These features are based on tweet text and Twitter user profiles. We adopted attributes that apply to the processing of natural Arabic language. It is worth noting that processing Arabic content is not straightforward since Arabic is a Semitic language. Semitic language properties that are still in their original phases introduce more issues to the processing of natural language compared with other languages [21]. One of these difficulties is that written diagrams are rarely used. Two words look identical without diacritics and are indistinguishable to machines. This decreases the efficiency of syntactic traits based on words [22]. Arabic is a language of inflection and derivation. This creates a serious problem for the removal of Arabic text and thus for the extraction of lexical attributes [23]. Proper nouns do not begin with a capital letter as they do in English, so extracting proper nouns from Arabic text is difficult [23].

Next, when the engineered features are generated, the system applies a feature selection technique to reduce the number of features, which reflect the performance of the classification algorithm. The main reason for using a feature selection technique rather than any other feature reduction technique is that our random forest classifier has its own methodology to rank features by their importance and select only the features that are important for the classification. Feature selection using fandom forest is classified as an embedded technique. Embedded techniques combine the capabilities of filter and wrapper methods. Algorithms that have their own built-in feature selection techniques implement them. Some of the advantages of embedded techniques are their extremely high accuracy and their generalizability and interpretability.

Lastly, the supervised learning data are divided into data sets for training and testing. The training data set is used to train the supervised classification method, which relies on the classification by random forest to form predictions for the study. On the other hand, the test data set is used to evaluate the trained models and determine performance in terms of accuracy, receiver operating characteristic (ROC), sensitivity, and specificity [24].

5. Data Collection

5.1. Constructing the Data Set

We used a targeting strategy to collect the data from the spamming Twitter accounts. The strategy follows a snowballing type data collection method, which means collecting the data from one user account then collect the data from all it added friend list. The reason behind choosing this strategy type is if we collect the data from an account that obtains rogue and spam contents there is a high chance the friends of this account have the same interest of obtaining these contents. Our strategy starts with a list of spamming accounts (e.g., 20 accounts) and then finds the list of friends for each account. We were required to determine the number of friends to find, and we chose 100 for each account. The process of finding the spammers' friend accounts was repeated three times to find the friends of friends for a specific spamming account. This strategy is more logical than any other method that collects spammers' data as well as their friends' data. Each account was collected by recording the 100 most recent tweets posted by the user. We started the data gathering process on 23 September and ended it on 23 October 2018. This resulted in a data set containing 10,096,919 tweets.

5.2. Labeling

Labeling tweets can be a tricky task, depending on the size of the data set. This process is required to apply supervised learning to the research problem. In this case, we observed all the collected tweets, collected the words that contain rogue and spamming contents, and stacked them in a list. We collected more than 100 words. Our approach to labeling all the collected tweets was to develop a simple programmed function that can find a particular word in a tweet. If a rogue or spamming word exists in a single tweet, then the function returns a true value. Otherwise, it returns a false value. We used this function to scan the tweets of each user account. If the total number of true values for an account

was greater than 60%, then the tweets from that account had mostly rogue and spamming contents, and the account was labeled a spam account.

6. Data Pre-processing

Pre-processing steps were applied to clean up the collected textual data and increase the efficiency of the decision-making system. The steps are given below.

1. All Twitter's separators such as '#' (hashtag symbol), 'RT' (Retweet indicator), '' (mention symbol), and double white spaces in the user's textual tweets were removed in order to work with clean text.
2. All the numbers were removed [1].
3. We removed all the stop words by deleting undervalued significant phrases, such as "a", "an", and "the"; this is a popular technique in the classification process.
4. All emoticons, symbols, pictographs, and flags in the text were removed.
5. All tweets that contained characters other than Arabic language characters were removed. Some users tweet or retweet in different languages (e.g., English), and our detection system concentrates only on Arabic tweets. After we applied this process, 62.09% of the tweets were removed.
6. Accounts that had fewer than 100 tweets were removed because our data collection process captured fewer tweets then the targeted number of tweets, and fewer tweets for an account meant less information as a basis for building the classification learner model. There were 52,788 user accounts with fewer than 100 tweets. Hence, around 46.40% of the total tweets needed to be removed.

After applying all the pre-processing steps, we ended up with a large-scale data set with about 2,909,455 tweets. The number of tweets was equivalent to 12,486 accounts, where 9333 accounts contained rogue and spam words, and 3153 accounts were legitimate. All the account tweets had more than 1000 clean texts to provide a reasonable amount of data to build the approach features, which affect the results of the classification algorithm.

7. Feature Engineering

This section describes the features extracted from each Twitter user account for spam detection purposes. It is possible to categorize the extracted features into user-based features and content-based features, both of which are described below.

7.1. User-Based Features

User-based features are related to the user's interactions, such as accounts followed by the user (referred to as friends) and those following the user (referred to as followers), or user activities, such as the times and frequencies of the user's tweets [2].

7.1.1. Account Reputation

On Twitter, users can build their social network by following friends and permitting other users to follow them. Spam accounts try to pay attention to large numbers of users. Twitter's spam and abuse policy indicates that "if you have a small number of followers compared to the amount of people you are following", then it may be considered a spam account. The reputation score is the number of followers divided by the total number of people in the user's network. The reputation of a user is defined as

$$\frac{Followers(j)}{Followers(j) + Followings(j)} \tag{1}$$

where $Followers(j)$ represents the number of accounts that follow user j, and $Followings(j)$ represents the number of friends ("following") that user j follows.

7.1.2. Account Age

Spammers create multiple Twitter accounts and move between them for a period of time to spread polluting content. Therefore, legitimate accounts have greater longevity than spam accounts [1].

7.1.3. Account Activity Per Day

Some spamming accounts avoid being detected by the account age feature by staying comparatively idle for an extended time after the account is created or engaging in recent spamming activities using a compromised account. Therefore, we combined the account age with certain account activities, such as the number of tweets that the user liked during the lifetime of the account, the number of retweets, the current number of followers of the account, the number of public lists of which the user is a member, and the number of follow-ups and number of tweets (including retweets) issued by the user. A legitimate user has more opportunities to create friends or followers per day than a spammer attempting to distribute spam tweets on Twitter. We created six features based on account activity per day.

7.2. Content-Based Features

Content-based features are the attributes of each tweet's content, including the text, hashes, mentions, and links. In this portion, we also compute the statistical measures of tweet entities. These features are described in the following subsection.

7.2.1. The Repeated Unique Entity

We use the term "entity" to indicate the tweet's unnatural language components, namely hashtags, URLs, and mentions. It is natural to expect spammers and non-spammers to differ in their use distributions of these entities. Therefore, to capture these distinctions, we created three features to count the most frequent elements.

Trending topics are the most-mentioned terms on Twitter at a given moment, in a week, or in a month. Users can tweet with a hashtag, with the # symbol followed by a word that describes or names the topics of interest. The term becomes a trending topic if there are many tweets containing the same term. Similarly, in their tweets, users can mention someone by using the symbol followed by the account name. Spammers often post a lot of unrelated tweets containing trending topics or mentioning famous accounts to attract legitimate users to read their tweets. Twitter considers an account to be spam "if a user posts multiple unrelated updates to a topic using the # symbol". Because spammers tend to repeat the same URL, hashtag, or mention in their tweets for the purpose of advertising and visibility, we calculated the total number of unique repeated entities. For instance, a normal user who is not subject-focused is anticipated to have fewer recurring distinctive URLs. On the other hand, a spammer tends to have a much higher number of repeated unique URLs. Please note that the unshortened URL is used because the shortened URL can be different each time the tweet is generated. We used the following expression to calculate the values, where *Entity* represents either a hashtag, URL, or mention, and i represents the tweets (of the most recent 100 tweets) that contain one of the mentioned entities.

$$\sum_{i=1}^{100} Unique(Entity) \tag{2}$$

7.2.2. Letter Counting

It is useful to count the number of letters for the following terms: username, tweet source (e.g., mobile or computer device), hashtag, URL, mention, the user-defined location, the user profile description, and the tweet text. Spammers use long names that often describe their purpose. For

example, a spammer that wishes to sell Twitter followers would choose a long name that contains all synonyms for that matter. We generated six features based on the text length [2].

7.2.3. Spam Tweet Ratio

Since we observed equivalent phrases and words in the content of spammers' tweets, we defined a metric to facilitate the identification of spammers. We developed a list of spam phrases that are frequently discovered in tweets from spammers. Over 100 words were included. Our predefined metric uses keyword information to find spam words in users' tweets. Then, it counts the tweets that include the spam words found in the most recent tweets for each user. Finally, it calculates the spam tweet ratio in the context of a user. For example, if we find a total of 40 spam tweets in the 100 most recent tweets, then the spam tweet ratio is 40/60, which equals 0.66. The expression of the spam tweet ratio is

$$\frac{Number\_of\_Spam\_Tweets}{Number\_of\_Non\_Spam\_Tweets} \quad (3)$$

7.2.4. Statistical Measurements

We obtained statistical measures such as the maximum, minimum, average, and total of the following: the number of the retweets, the number of tweets the user has liked in the account's lifetime, the current number of followers of the account, the number of accounts the user is following, the number of public lists of which the user is a member, and the number of tweets (including retweets) issued by the user. For example, we can compute the average number of accounts followed by a user by averaging this measure over the most recent 100 tweets for a particular user. Around 28 features are based on statistical measurements [2].

8. Feature Analysis

8.1. Feature Ranking

Choosing the correct data set attributes can be the difference between medium performance with a long training time and high performance with a short training time [2]. Besides classification, one of the advantages of the random forest classifier is its feature importance methodology, which uses a "Gini index" to assign a score and rank the features in the data [25]. It generates ratings that are referred to as the "Mean Decrease Gini" as a measure of significance to reveal the degree to which each attribute contributes to data homogeneity [26]. It operates as follows:

- The Gini index is calculated at the root node and at both leaves each time a feature is used to divide data at a node. The Gini index reflects homogeneity: it is 0 for all-homogeneous data and 1 for all-heterogeneous data.
- The distinction between the child nodes' Gini index and the dividing root node for the feature is calculated and standardized.
- It is also said that the nodes result in data 'purity', which implies that the data are categorized more easily and efficiently. If the purity is high, then there is a large mean reduction in the Gini index.
- Therefore, the mean decrease in the Gini index is the highest for the most important feature.
- Such features are useful for classifying data and, when used at a node, are likely to split the data into pure single-class nodes. Therefore, during splitting, they are used first.
- Therefore, for each feature, the general mean decrease in Gini importance is calculated as the percentage of the sum of the number of splits in all trees that include the function to the number of samples it divides.

Figure 4 shows the top ten most important features in our data set by implementing the 10-fold cross-validation method, repeated two times [27]. The most important feature is the text ratio of the

spam words, with a mean decrease Gini of around 700, which is convincing to have a high result since it is a text-based feature.

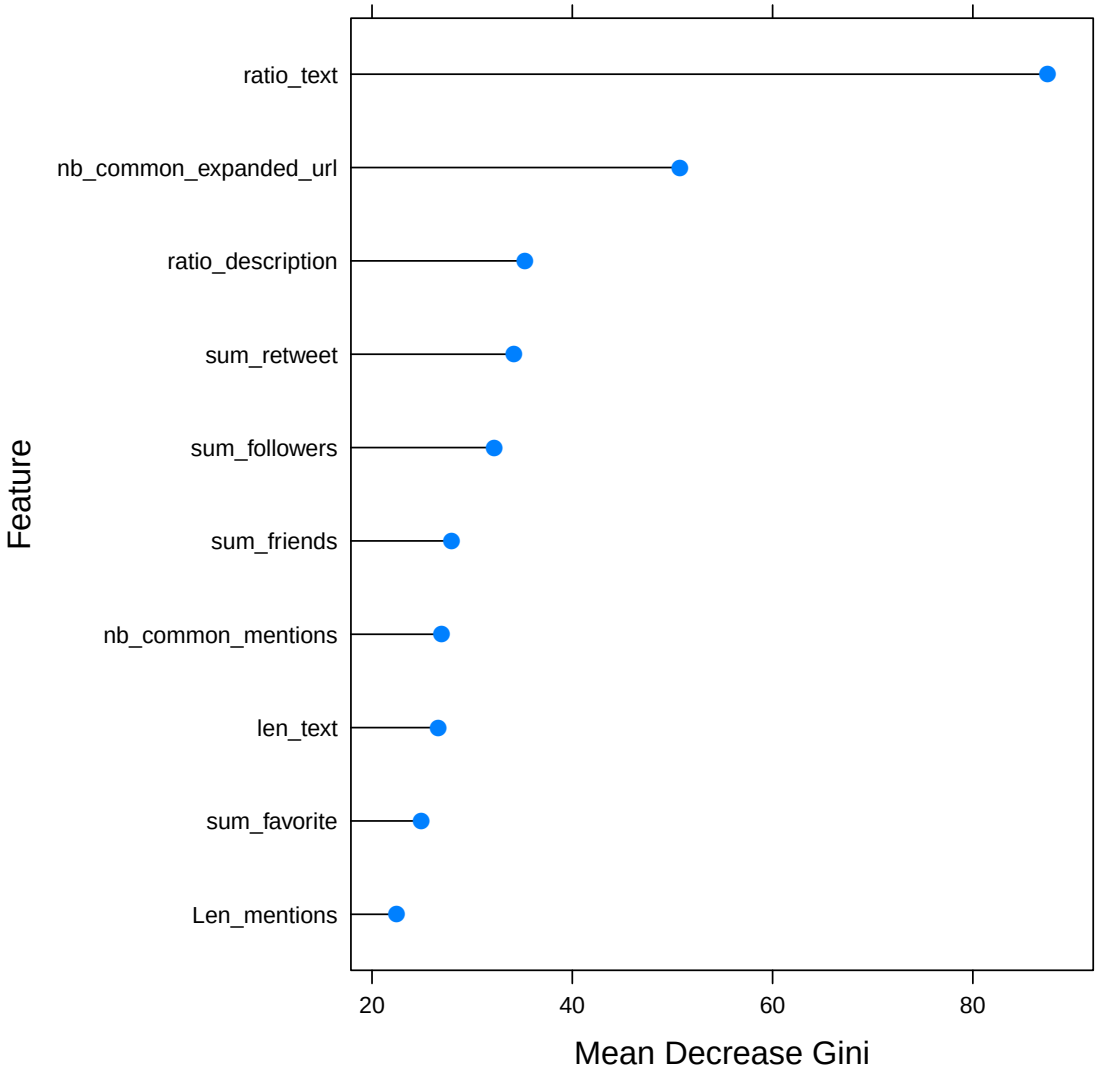

Figure 4. The top ten most important features.

8.2. Feature Relationship

We used the Pearson parametric correlation technique to define the correlation coefficients to better understand the relationships and determine which features provide invisible data [28]. Figure 5 presents the correlation coefficients in a color-coded plot for all pairs of features. White squares show that there is no correlation between the function pair, blue indicates a positive correlation, and red implies a negative correlation. The legend color scale ranges from -1 to 1 at the bottom of the matrix plot and corresponds to the correlation coefficient: the darker the color, the greater the coefficient of correlation between the pairs of features [29]. Indeed, this plot defines the hidden structures and patterns that might occur between the features and guides our choice of features since high-scoring features can improve the classification algorithm's efficiency with fewer features.

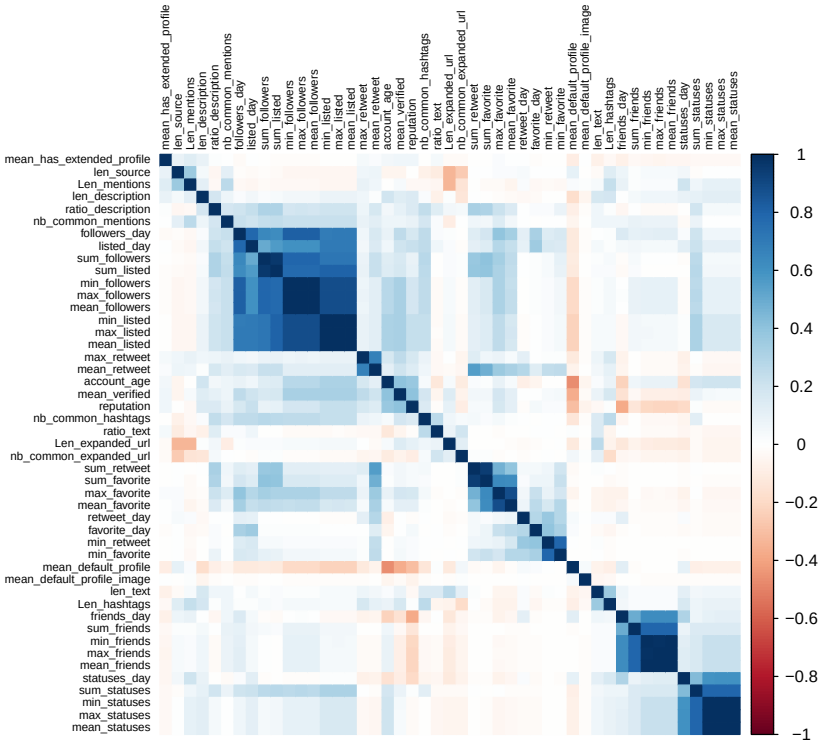

Figure 5. Correlation matrix for all features.

9. Supervised Classification

Random forest is a supervised learning algorithm that can be used for classification as well as regression. As its name suggests, random forest comprises a large number of individual decision trees that function as an ensemble [30]. Each tree spreads a class prediction in the random forest, and the class with the most votes forms the predictions in our model, as shown in Figure 6.

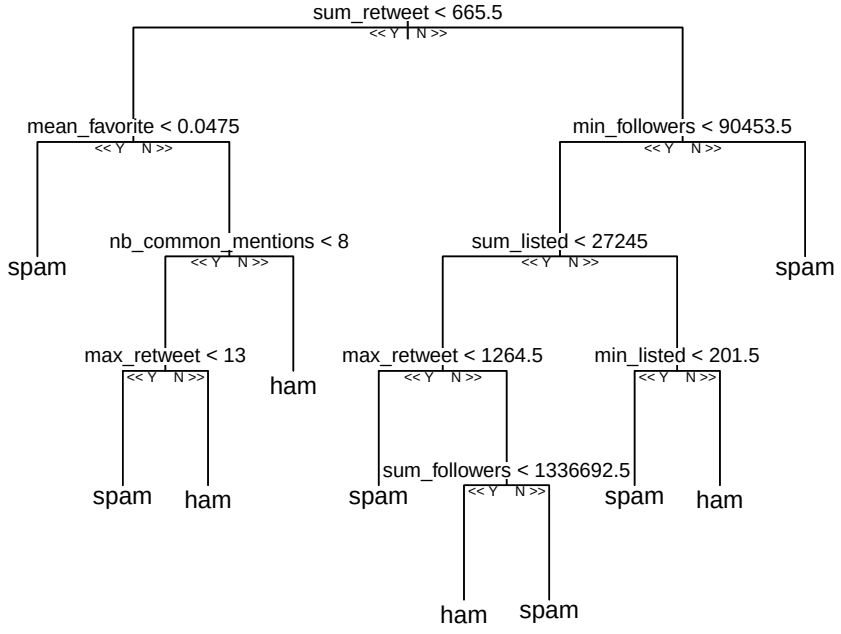

Figure 6. Random forest tree visualization.

RF is a straightforward and strong classifier [31] because an ensemble of a large number of comparatively uncorrelated models (trees) working as a committee outperforms any of the individual

constituent models [32]. A low correlation between models is key. Just as investments with low correlations (such as stocks and bonds) come together to form a portfolio that is larger than the sum of its parts, uncorrelated models can produce more accurate set predictions than any of the individual predictions. The reason for this fantastic effect is that trees safeguard each other from their individual errors (as long as errors are not constantly being made in the same direction) [33]. Although some trees may be incorrect, many other trees are going to be right; hence, the trees can move in the right direction as a group. The prerequisites for a well-performing RF classifier are that (i) there must be some real signal in our features so that models built using these features have results that are better than random guessing, and (ii) the predictions (and therefore the errors) made by the individual trees need to have low correlations with each other [30].

10. Experimental Result

The findings of our experiments are based on several metrics of classification. The accuracy metric is the percentage divided by all observations of correctly classified observations. Biometric performance metrics were also acquired. These results are based on the receiver operating characteristic (ROC) curve. The ROC illustrates the classification model's capacity for distinguishing between positive and negative classes. The ROC depends on two performance measurements: sensitivity and specificity. Sensitivity measures the number of positive (first) class samples that are correctly predicted. Specificity measures the number of negative (second) class samples that are correctly predicted. An ideal classification model that has high sensitivity, specificity, and ROC metric rates is 100% if the model generates all predictions completely. If the model results in rates near 50%, then it is no better than an arbitrary guess [34].

10.1. Feature Selection Results

Methods of feature selection can be used to create many models with distinct subsets of a data set and to distinguish variables that are not needed to create an effective model [2]. A common simple technique for feature selection is provided by the so-called recursive feature elimination (RFE) method [35]. In the RFE method, a model is built with all the variables, and then the algorithm removes the weakest features one by one until it reaches the number of features specified. We need to indicate the number of attributes to use when using RFE. However, this is often not known at first. Cross-validation can identify the ideal number of attributes. In this experiment, we implemented the RFE method with our data set, and the accuracy rates were based on the 10-fold cross-validation strategy, repeated three times [27]. Each iteration used the random forest algorithm to measure the model. The algorithm was set up to investigate all possible attribute subsets. All 47 features were selected in this experiment [33]. Figure 7 shows the accuracy rates of the different feature subset sizes: it started at a low level of performance (e.g., 0.9046, 0.8378), and then the rates increased incrementally as the number of features increased. The highest accuracy rates resulted in 16 of the features with a 0.9321 accuracy rate. Later, the performance of the classifier stabilized between 0.9298 and 0.9306. From these results, we deduced that just 16 features yield almost comparable results.

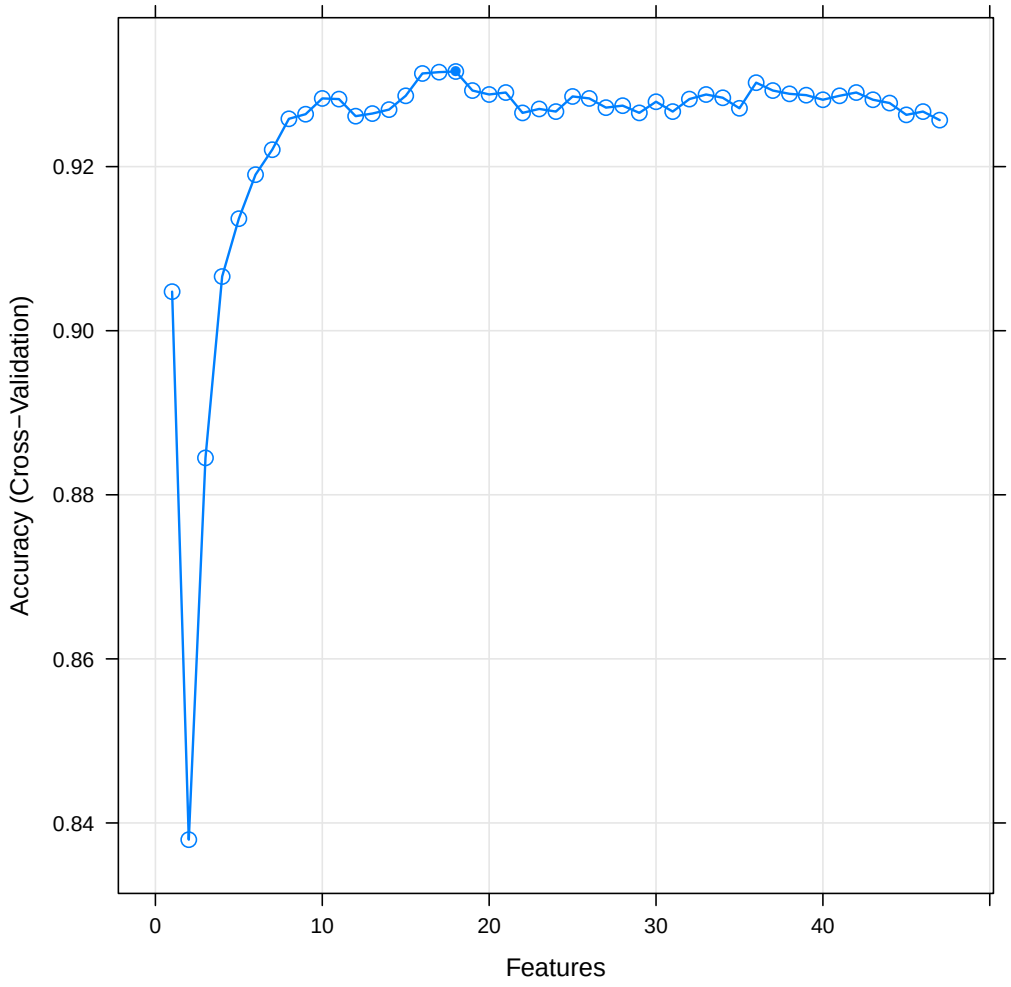

Figure 7. Feature selection results.

10.2. Tuning Classification Algorithm Parameters

Tuning the algorithm's parameters is a useful exercise to search for the best combinations of parameters to discover whether our proposed approach is optimal [36]. However, searching for algorithm parameters can be difficult since there are many options to examine. In this case study, we evaluated the tuning of some important parameters and searched for a method that could locate good algorithm parameters [33]. We tuned two parameters—namely the number of variables randomly sampled as candidates at each split (mtry) and the number of trees to grow (ntree)—to test the effect on our random forest model. There are many other parameters, but these two are perhaps the most likely to have the biggest effect on our final accuracy. We implemented 10-fold cross-validation with three repeats in all our experiments [27].

10.2.1. Random Search

One search approach is the use of random values within a range. This approach is used because we are uncertain of the value, we want to overcome any biases, and we might have to set the parameter [37]. Figure 8 shows that the highest ROC value for mtry was 8 with an accuracy of 0.9819.

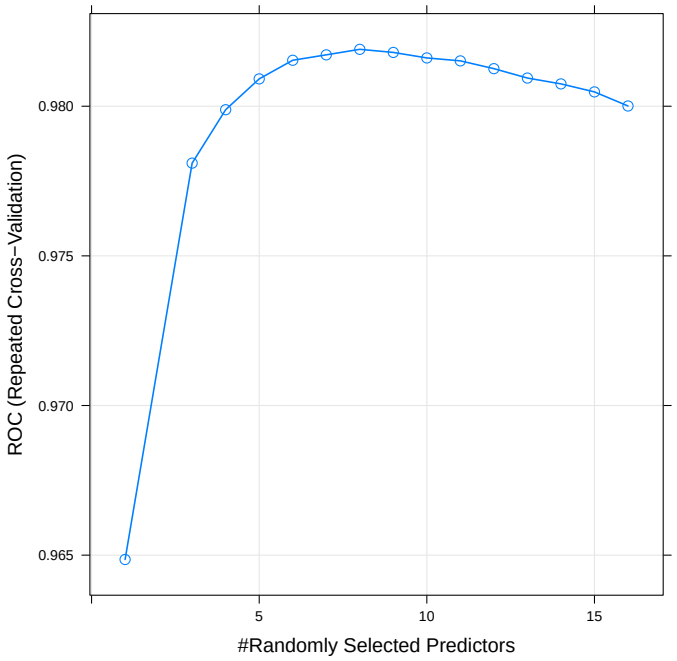

Figure 8. ROC results of the random search method.

10.2.2. Grid Search

Another search method is to attempt to identify an algorithm grid. Each grid axis is an algorithm parameter, and the grid points are particular parameter combinations. Because we only tune one parameter, the grid search is a linear search of candidate values through a vector [37]. Figure 9 shows that the most accurate value for mtry was 8 with an ROC of 0.9818.

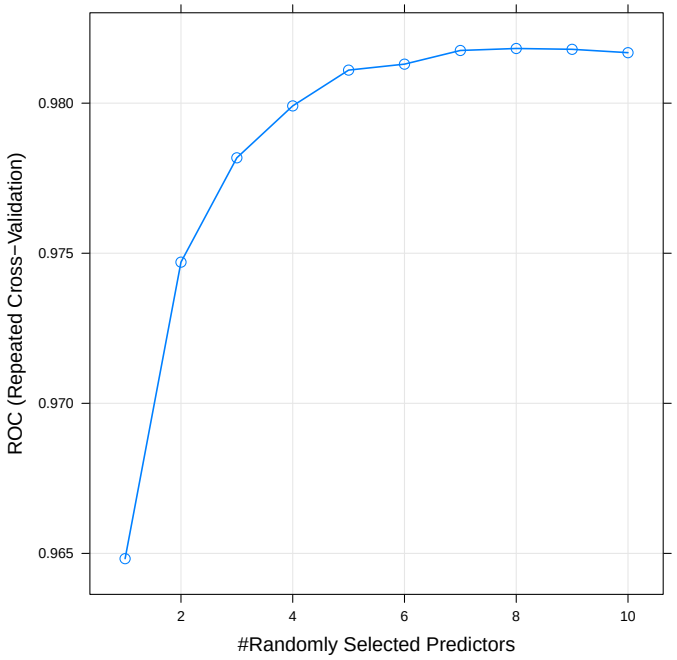

Figure 9. ROC results of the grid search method.

10.2.3. Search by the Number of Trees

Another strategy is to manually generate many models for our RF classifier algorithm and transfer them straight to the algorithm in separate parameters [36]. In this experiment, we evaluated the same classification algorithm but modified so that it supported tuning both the mtry and ntree parameters. We defined the algorithm to use by identifying a list containing the number of custom-named components that the algorithm seeks, such as how to fit and predict. Figure 10 shows that the most accurate values for ntree and mtry were 2500 and 8, with an ROC rate of 0.9820. We may have noticed some effects of communication between the set of trees and the ntree value. Nonetheless, if we had chosen the best value for mtry of 8 using a random and grid search and the highest ntree value of 2500 using grid search, then, in this scenario, we would have reached the same number for tuning as that measured in this infused search. This is a great confirmation.

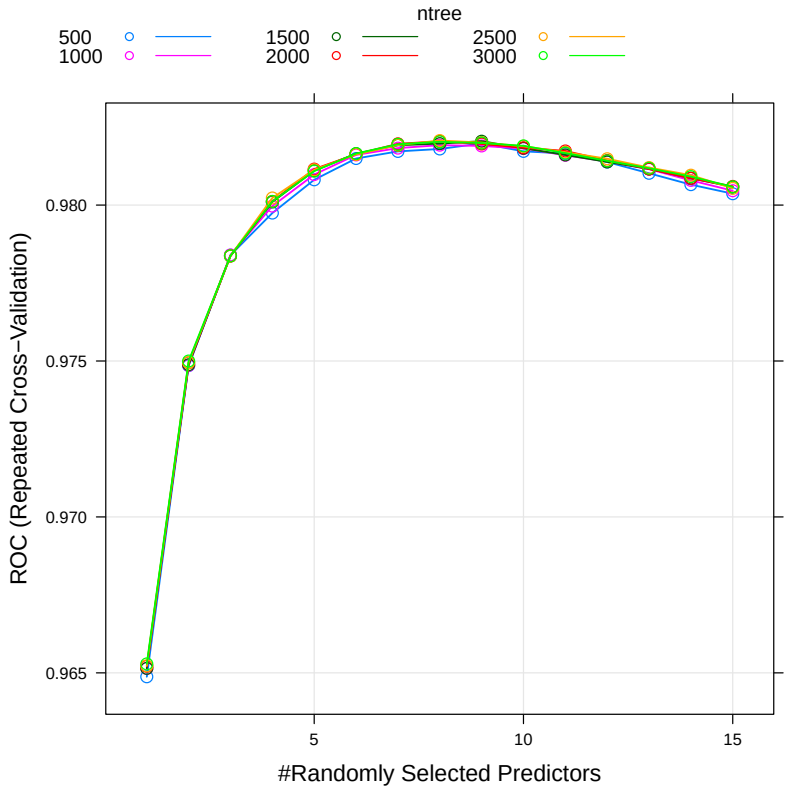

Figure 10. ROC results of searching by the number of trees.

10.3. Performance Results for a Different Number of Users

To gain a better understanding of the effect of the number of users on the performance of our proposed approach, we evaluated it for a real case scenario using a known training and testing validation strategy in which the classifier is influenced by increasing the number of subjects used for training, and the rest of the subjects are used for testing [24]. Using this strategy for our experiment, we separated the data set according to different numbers of users to train the classifier, with the rest of the data used to test it. The division of data started from 100 and was incremented by 100 for each run until reaching 12,400 users. We calculated the accuracy, sensitivity, and specificity of the classifier for each run. Moreover, all the classifier parameters were fixed for each run; for example, the number of variables randomly sampled as candidates at each split was 8, as used in our previous experiment. For each increasing number of subjects, we performed 5-fold cross-validation [27] to produce the results. Figure 11 shows that the accuracy, sensitivity, and specificity rates of the RF classifier gradually increased as we added more users to the training step. For instance, when the classifier was trained on 100 subjects (12,386 subjects used for the testing set), the accuracy, sensitivity, and specificity rates

were 0.7605, 0.3165, and 0.9115, respectively. Then, the RF classifier obtained higher resulting rates as we increased the number of training subjects.

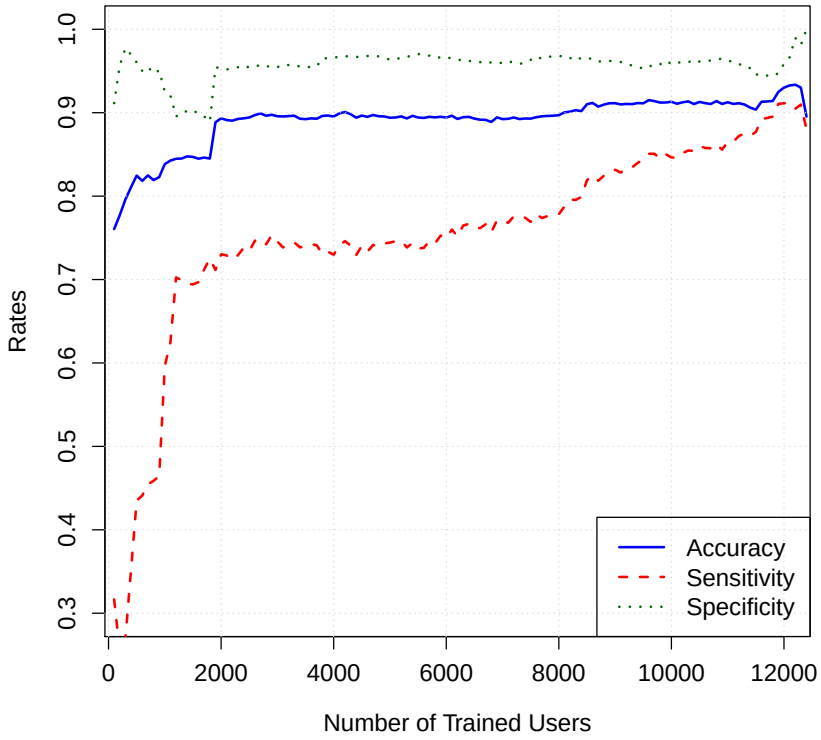

Figure 11. Performance results for different numbers of users.

11. Conclusions and Future Work

In this paper, we address the issue of using text classification to detect rogue and spam accounts with Arabic tweets. We offer a specific contribution to the characterization and modeling of Arab spammers on Twitter, a topic that represents an urgent financial, political, and computational issue [2]. We discovered that the populations of English and Arabic spammers vary considerably, making any spam detection model unsuitable for one population. Moreover, the evasion methods of spammers evolve intrinsically. They modify their behavior continually, drifting away from their old behavioral models and rendering old detection systems ineffective. In this work, we collected a large-scale tweet data set. We describe the spammers' evasion methods and how detection features can be tailored to take into account the consequences of evasion. Using these features for our approach, we achieved high performance by implementing one of the well-known classification algorithms, random forest. We analyzed the 47 features generated and selected the 16 most significant ones. We also maximized the accuracy, sensitivity, and specificity rates by tuning the proposed classifier.

In our future work, we plan to refine our results by expanding the data set. We also intend to further explore this topic, particularly in terms of detecting and characterizing spam content, methods, and campaigns (as opposed to spamming accounts) [38]. In addition, we will try to distinguish translated tweets created by an Arab writer [39]. We may attempt to construct a framework with more features to identify rogue and spam accounts in other languages.

Author Contributions: The authors contributed equally to this work.

Funding: This work was supported by Deanship of Scientific Research in University of Tabuk. The fund number is S-0261-1439.

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

References

1. Abozinadah, E.A.; Mbaziira, A.V.; Jones, J. Detection of abusive accounts with Arabic tweets. *Int. J. Knowl. Eng.-IACSIT* **2015**, *1*, 113–119. [CrossRef]
2. El-Mawass, N.; Alaboodi, S. Detecting Arabic spammers and content polluters on Twitter. In Proceedings of the 2016 Sixth International Conference on Digital Information Processing and Communications (ICDIPC), Beirut, Lebanon, 21–23 April 2016; pp. 53–58.
3. Abdurabb, K. Saudi Arabia has highest number of active Twitter users in the Arab world. *Arab News*, 27 June 2014.
4. Mari, M. Twitter usage is booming in Saudi Arabia. *GlobalWebIndex (Blog)* **2013**, *20*. Available online: https://blog.globalwebindex.com/chart-of-the-day/twitter-usage-is-booming-in-saudi-arabia/ (accessed on 29 October 2019).
5. Benevenuto, F.; Magno, G.; Rodrigues, T.; Almeida, V. Detecting spammers on twitter. In *Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference (CEAS)*; CiteSeerx: Princeton, NJ, USA, 2010; Volume 6, p. 12.
6. Mccord, M.; Chuah, M. Spam detection on twitter using traditional classifiers. In *International Conference on Autonomic and Trusted Computing*; Springer: Berlin, Germany, 2011; pp. 175–186.
7. Wang, A.H. Don't follow me: Spam detection in twitter. In Proceedings of the 2010 International Conference on Security and Cryptography (SECRYPT), Athens, Greece, 26–28 July 2010; pp. 1–10.
8. Alhumoud, S.O.; Altuwaijri, M.I.; Albuhairi, T.M.; Alohaideb, W.M. Survey on arabic sentiment analysis in twitter. *Int. Sci. Index* **2015**, *9*, 364–368.
9. Chaabane, A.; Chen, T.; Cunche, M.; De Cristofaro, E.; Friedman, A.; Kaafar, M.A. Censorship in the wild: Analyzing Internet filtering in Syria. In Proceedings of the 2014 Conference on Internet Measurement Conference, Vancouver, BC, Canada, 5–7 November 2014, ACM: New York, NY, USA: 2014; pp. 285–298.
10. El-Mawass, N.; Alaboodi, S. Data Quality Challenges in Social Spam Research. *J. Data Inf. Qual.* **2017**, *9*, 4. [CrossRef]
11. Najafabadi, M.M.; Domanski, R.J. Hacktivism and distributed hashtag spoiling on Twitter: Tales of the IranTalks. *First Monday* **2018**, *23*. [CrossRef]
12. Ameen, A.K.; Kaya, B. Detecting spammers in twitter network. *Int. J. Appl. Math. Electron. Comput.* **2017**, *5*, 71–75. [CrossRef]
13. Al-Rubaiee, H.; Qiu, R.; Alomar, K.; Li, D. Sentiment analysis of Arabic tweets in e-learning. *J. Comput. Sci.* **2016**,*11*, 553–563. [CrossRef]
14. Sriram, B.; Fuhry, D.; Demir, E.; Ferhatosmanoglu, H.; Demirbas, M. Short text classification in twitter to improve information filtering. In Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Geneva, Switzerland, 19–23 July 2019, ACM: New York, NY, USA, 2010; pp. 841–842.
15. Mubarak, H.; Darwish, K.; Magdy, W. Abusive language detection on Arabic social media. In Proceedings of the First Workshop on Abusive Language Online, Vancouver, BC, Canada, 4–7 August 2017; pp. 52–56.
16. Alshehri, A.; Nagoudi, A.; Hassan, A.; Abdul-Mageed, M. Think before your click: Data and models for adult content in arabic twitter. In Proceedings of the 2nd Text Analytics for Cybersecurity and Online Safety (TA-COS-2018), 2018. Available online: https://pdfs.semanticscholar.org/0515/b46e219b2ea6e7f843e42e79ed2cf5591b61.pdf (accesesed on 29 October 2019).
17. Al-Eidan, R.M.B.; Al-Khalifa, H.S.; Al-Salman, A.S. Measuring the credibility of Arabic text content in Twitter. In Proceedings of the 2010 Fifth International Conference on Digital Information Management (ICDIM), Thunder Bay, ON, Canada, 5–8 July 2010; pp. 285–291.
18. Rsheed, N.A.; Khan, M.B. Predicting the popularity of trending arabic news on twitter. In Proceedings of the 6th International Conference on Management of Emergent Digital EcoSystems, Buraidah, Al Qassim, Saudi Arabia, 15–17 September 2014; pp. 15–19.
19. Vijayarani, S.; Janani, R. Text mining: open source tokenization tools—An analysis. *Adv. Comput. Intell.* **2016**, *3*, 37–47.
20. Perera, R.D.; Anand, S.; Subbalakshmi, K.; Chandramouli, R. Twitter analytics: Architecture, tools and analysis. In Proceedings of the 2010-MILCOM 2010 Military Communications Conference, San Jose, CA, USA, 31 October–3 November 2010; pp. 2186–2191.

21. Haidar, B.; Chamoun, M.; Serhrouchni, A. A multilingual system for cyberbullying detection: Arabic content detection using machine learning. *Adv. Sci. Technol. Eng. Syst. J.* **2017**, *2*, 275–284. [CrossRef]
22. Abbasi, A.; Chen, H. Applying authorship analysis to Arabic web content. In *International Conference on Intelligence and Security Informatics*; Springer: Berlin, Germany, 2005; pp. 183–197.
23. Al-Shammari, E.T.; Lin, J. Towards an error-free Arabic stemming. In Proceedings of the 2nd ACM Workshop on Improving Non English Web Searching, Napa Valley, CA, USA, 30 October 2008; pp. 9–16.
24. Xie, X.; Ho, J.W.; Murphy, C.; Kaiser, G.; Xu, B.; Chen, T.Y. Testing and validating machine learning classifiers by metamorphic testing. *J. Syst. Softw.* **2011**, *84*, 544–558. [CrossRef] [PubMed]
25. Sandri, M.; Zuccolotto, P. A bias correction algorithm for the Gini variable importance measure in classification trees. *J. Comput. Graph. Stat.* **2008**, *17*, 611–628. [CrossRef]
26. Louppe, G.; Wehenkel, L.; Sutera, A.; Geurts, P. Understanding variable importances in forests of randomized trees. In Proceedings of the 26th International Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; pp.431–439.
27. Kohavi, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. *InIjcai* **1995**, *14*, 1137–1145.
28. Sedgwick, P. Pearson's correlation coefficient. *BMJ* **2012**, *345*, e4483. [CrossRef]
29. Chen, P.Y.; Smithson, M.; Popovich, P.M. *Correlation: Parametric and Nonparametric Measures*; No. 139; Sage: New York, NY, USA, 2002.
30. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]
31. Wainberg, M.; Alipanahi, B.; Frey, B.J. Are random forests truly the best classifiers? *J. Mach. Learn. Res.* **2016**, *17*, 3837–3841.
32. Brownlee, J. Machine Learning Mastery. Available online: http://machinelearningmastery.com/discover-feature-engineering-howtoengineer-features-and-how-to-getgood-at-it (accessed on 29 October 2019).
33. Kuhn, M. Building predictive models in R using the caret package. *J. Stat. Softw.* **2008**, *28*, 1–26. [CrossRef]
34. Sokolova, M.; Japkowicz, N.; Szpakowicz, S. Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation. In *Australasian Joint Conference on Artificial Intelligence*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 1015–1021.
35. Granitto, P.M.; Furlanello, C.; Biasioli, F.; Gasperi, F. Recursive feature elimination with random forest for PTR-MS analysis of agroindustrial products. *Chemom. Intell. Lab. Syst.* **2006**, *83*, 83–90. [CrossRef]
36. Robnik-Šikonja, M. Improving random forests. In *European Conference on Machine Learning*; Springer: Berlin/Heidelberg, Germany, 2004; pp. 359–370.
37. Sonobe, R.; Tani, H.; Wang, X.; Kobayashi, N.; Shimamura, H. Parameter tuning in the support vector machine and random forest and their performances in cross-and same-year crop classification using TerraSAR-X. *Int. J. Remote. Sens.* **2014**, *35*, 7898–7909. [CrossRef]
38. Jamal, N.; Xianqiao, C.; Aldabbas, H. Deep Learning-Based Sentimental Analysis for Large-Scale Imbalanced Twitter Data. *Future Internet* **2019**, *11*, 190. [CrossRef]
39. Imam, N.; Issac, B.; Jacob, S.M. A Semi-Supervised Learning Approach for Tackling Twitter Spam Drift. *Int. J. Comput. Intell. Appl.* **2019**, *18*, 1950010. [CrossRef]