doi: 10.20532/cit.2021.1005221

Identifying Spam Activity on Public Facebook Pages

Hakim Azri, Hafida Belbachir and Fatiha Guerroudji Meddah

Université des Sciences et de la Technologie d'Oran - Mohamed Boudiaf – Department of Computer Science, Algeria

Since their emergence, online social networks (OSNs) keep gaining popularity. However, many related problems have also arisen, such as the use of fake accounts for malicious activities. In this paper, we focus on identifying spammers among users that are active on public Facebook pages. We are specifically interested in identifying groups of spammers sharing similar URLs. For this purpose, we built an initial dataset based on all the content that has been posted upon feed posts on a set of public Facebook pages with high numbers of subscribers. We assumed that such public pages, with hundreds of thousands of subscribers and revolving around a common attractive topic, make an ideal ground for spamming activity. Our first contribution in this paper is a reliable methodology that helps in identifying potential spammer and non-spammer accounts that are likely to be tagged as, respectively, spammers/non-spammers upon manual verification. For that aim, we used a set of features characterizing spam activity with a scoring method. This methodology, combined with manual human validation, successfully allowed us to build a dataset of spammers and non-spammers. Our second contribution is the analysis of the identified spammer accounts. We found that these accounts do not display any community-like behavior as they rarely interact with each other, and are slightly more active than non-spammers during late-night hours, while slightly less active during daytime hours. Finally, our third contribution is the proposal of a clustering approach that successfully detected 16 groups of spammers in the form of clusters of spam accounts sharing similar URLs.

ACM CCS (2012) Classification: Information systems → World Wide Web → Web applications → Social Networks

Information systems → Information retrieval

Information systems → Information systems applications → Data mining → Clustering

Keywords: online social networks, fake accounts, spam, clustering

1. Introduction

Recently, social networks are experiencing rapid expansion, with a constantly increasing number of users usually looking for meaningful connections. Facebook is an open and decentralized OSN and the largest one in the world with its 2 billion monthly active users [1]. It is also a user-centered platform, where the user is allowed to build a virtual identity that is rich with information. However, these properties have also encouraged the use of fake identities, mostly for malicious purposes.

Using fake accounts as a means to spread spam is one particular case that is having a negative impact on the Facebook's user experience and security. Even more critical is when an entity (person or organization) creates and manipulates groups of fake accounts at large scales. Such malicious entities, for instance, may use groups of fake accounts to inflate pages with fake likes and/or followers [2], a practice that is, in fact, considered by social media experts as worse for those pages than keeping a low number of likes [3]. Some spamming cyber-attackers even promised fake likes via spam botnets to users who provided them with their app's access token, which they used to spread spam through larger audiences [4]. Such attacks may also aim for manipulating public opinion by spreading spam [5]. Spam attacks that are launched at large-scales, using multiple accounts, are generally referred to as *spam campaigns*, and may have a serious impact on many levels.

Up till now, Facebook has been dealing with the problem of spammers and fake account activity.

As a result, various strategies were being adopted by the corporation [6]. Like most of the other OSNs, allowing users to report any suspicious content to Facebook's team is the simplest strategy available. However, this feature is also misused sometimes, as some users get often reported as spammers themselves when they post content in favor of a certain opinion on which other users might disagree [7].

At a more sophisticated level, the company is using an automated detection to identify spam and fake accounts. During a recent purging operation [8], around 30,000 fake accounts have been detected in France, showing a repeated posting of the same content or an increase in the posted messages. Facebook's anti-spamming algorithm is also continuously evolving. The corporation announced recently that the algorithm has been changed to curb tiny groups of spammers who share vast amounts of low-quality public posts daily, with over 50 posts a day [9]. However, some measures that Facebook applies might also be disadvantaging to some users or pages: while content from popular authentic accounts on the network may get shared at a quick pace, such content is often assumed to be spam by Facebook if an account is not verified, and then it is throttled for a period of time [10].

Few works dealt with the problem of multiple fake identities within spam campaigns on Facebook. There are no large scale studies on Facebook for analyzing spam campaigns. This is most probably due to the difficulty of building ground truth datasets by researchers, mainly because of Facebook's policy regarding collecting user data.

In this work, our aim is the identification of spammers and groups of spammers using a set of public Facebook pages with high numbers of subscribers, since we assume it is ideal for spammers to target users on pages. Our contributions in this work can be summarized, as follows:

- We propose a methodology to build a dataset of spammer/non spammer accounts on Facebook that is based on human verification, but which facilitates the process of manually tagging these accounts. The methodology allows us to save time and effort by avoiding random sampling of the accounts for their manual tagging, which could have taken an enormous time, especially to find the desired number of spammers within the group of 600,000 users. The methodology uses a set of features characterizing the likelihood of spam activity with a scoring method that scores each account. Combined with manual human tagging, it allowed us to build a dataset comprised of 1000 spammer accounts and 1000 non-spammers in a reasonable amount of time.
- We perform a comparison of the temporal activity patterns of spammers with those of non-spammers. We also analyze how the validated 1000 spammers interacted with each other by tracing a graph where vertices represent spammers and each edge represents an interaction between spammers through tagging, commenting, or liking content. Our aim was to discover any potential community-like behavior between spammer accounts.
- We propose a URL-similarity approach for spam group detection and apply it on a subset of users from the previous dataset in order to identify groups of spammers, where users are linked within a graph if they share similar URLs. Our goal was to identify any clusters of spammers that might be manipulated by the same entities or for the same purposes, thus, these accounts most likely use not only identical URLs, but probably also ones that have some similarity between them.

The remainder of this paper is organized, as follows: Section 2 presents various approaches from the related work that dealt with spam accounts on OSNs. In Section 3, we present the assumptions behind our work and our methodology for collecting data. Section 4 presents our method for identifying spammer and non-spammer accounts on Facebook, along with our findings from analyzing the identified spammer dataset. Section 5 details our URL similarity-based approach for detecting clusters of spammers and the experimentation results. Finally, Section 6 presents our conclusions and future work.

2. Related Work

A huge amount of work has been dedicated to spam detection focusing on social media and on other platforms (*e.g.*, email). On social media, the problem is more of identifying fake accounts along with the abnormal behavior they might show, since these are often used as a vehicle to spread spam. Therefore, various characteristics have been studied for the detection of spammers and fake accounts on OSNs, such as the frequency of posting content, content similarity, the malicious use of apps and the excessive use of URLs [11, 12, 13]. Additionally, malicious social bots, usually used to spam on OSNs, have been also studied based on their odd, automatically generated behavior [14, 15]. Particularly in [15], it has been shown that humans on Twitter tend to post more than bots, although bots have specific periods of activity where their posting rate is much greater than that of real humans. It was also found that some temporal patterns such as the elapsed time between each couple of successive posts, or the temporal patterns for the connection sessions to the OSN may be relevant for the detection.

Moreover, some of the previous works rely on friendship links to detect spammers and fake accounts, specifically exploiting the difficulty for fake accounts to establish mutual relationships with honest users. In [16], this property was expressed through the *follower ratio* for Twitter users, which is the ratio between the number of followers and the number of followings. Also, in [17], the authors used the accepted friendship request rates, such as fake users showing a high acceptance rate for the incoming requests they received from others compared to honest users, and a low acceptance rate for their outgoing friendship requests towards other users. Some detectors have also used topology features of the social connections graph, such as the *clustering coefficient* of a vertex, the *betweenness*, the *assortativity*, the *Jaccard coefficient*, *etc.* [18, 19].

It is crucial to point out that fake account and spammer detection techniques and methods have evolved throughout time, mainly in response to malicious entities leveraging their attacks to improve their targeting in quality and scale, and to avoid detection. Temporal and textual patterns can indeed help researchers and companies, especially with the detection of automated accounts [15]. However, modern bots avoid such patterns and are sophisticated enough to simulate the human behavior [20]. Furthermore, certain fakes are set to "clone" other profiles' content to emulate the image of an honest account. Others can modify the content they publish so they escape the detection, such as intentionally using typographical errors (*e.g.*, "C lick" instead of "Click"). As for demographic features (*e.g.*, age of a user), they may be ineffective in most cases due to the unpredictability of information that a user usually gives, which is a fact that attackers can take to their advantage.

As our work aims to identify groups of spammers, *i.e.*, spam campaigns, we were especially interested in the related work that addresses this problem. Many approaches focus on the detection of Sybil attacks, which is the case where a large group of fake accounts called Sybils is created by one entity to launch collaborative attacks [21]. Therefore, spam campaigns can also be considered as *Sybil attacks*. Most of the proposed solutions in this area rely completely on the topology of the social graph (relationships between users) [21, 22, 23] and focus on finding vertices (users) with few social connections. Some of these approaches have also used machine learning to improve the efficiency of the graph-based detection [24].

In our work, we do not rely on such schemes. Other researchers do not necessarily propose graph-based Sybil detection schemes to identify groups of malicious users, but instead try to detect any existing group behavior characterizing a collaborative nature for these users, such as similar patterns when uploading or liking content, or following other users at around the same time [25, 26]. In [27], the authors rather focused on the *clickstream* information of a user to classify them, because clickstreams are traces of click-through events generated by online users during each web browsing session.

The efforts of spammer detection on Facebook include the work done in [28], where spammers were identified on Facebook pages within a graph where connected users, represented as vertices, share similar patterns regarding the frequency of posting on their walls, page likes and URL sharing. Slightly similar approaches have been proposed in [29] and [30]. In [30], the authors aimed at grouping similar URLs that can be captured by one sufficiently specific regular expression, allowing the detection of groups of spammers as clusters of users.

As spammers improve their strategies on Facebook, they also tend to post their malicious content directly on public Facebook pages, not only because it relieves them from the necessity to establish trust relationships with honest users in order to target them (with spam links generally), but this also allows them to target, on public pages, users that share common specific characteristics and interests, improving thus the quality of their target communities in accordance to their goals.

3. Background

3.1. Hypothesis

For our proposed work, we assumed that spammers on Facebook tend to target public pages. This relieves the attackers from the need to form direct social relationships with the authentic users to reach them, which might seem like a burden to fake accounts, especially on a user-centered OSN like Facebook, where relationships are based on trust and require mutual validation. Also, we assume that popular pages, *i.e.*, pages with very high numbers of subscribers (more than 100,000), that focus on a specific topic (*e.g.*, fitness), attract users who share common interests. We assume that such conditions make it easier for spammers to target users and increase the visibility of their spam, therefore creating an ideal environment for them to build their nests [31].

3.2. Data Collection

Since it is unlikely to find any Facebook spam datasets that include user-generated content online, we decided to build our own dataset by proposing our own methodology. The idea is to use the content (comments) generated by users on public pages in order to build a ground-truth dataset with a set of spammer (fake) user accounts and a set of non-spammer (authentic) user accounts. Therefore, the first step was collecting the comments that were posted on several public Facebook pages.

3.2.1. Challenges During Data Collection

One of the most difficult steps during our work was the data collection part. We chose to work with the Facebook API over other methods for crawling data, which might be considered illegal if launched on a large scale [32]. However, Facebook imposes strict rules regarding the API use and data gathering due to privacy concerns.

Even though most personal information on a user and their friend list can be publicly accessed via the website if the user is allowing it, it cannot be requested through the API unless permission is obtained from the user in the form of a *user token*. Besides, there is an API call rate limit that is imposed, with a user only able to send a maximum of 200 calls within any given 60-minute time window [33]. Also during our data collection phase, there were often unpredictable errors on the Facebook server's end that we couldn't identify, resulting in blocking our API calls.

3.2.2. Initial Dataset Creation

We chose a set of 15 public Facebook pages, each having more than 100,000 subscribers, and all of them sharing a common topic: fitness. We only took into consideration pages that publish content in English. After a first glance at the comments posted to these pages, we noticed a considerable amount of spam activity.

We used Facebook Graph API to collect all the public feed posts from the pages during March of 2015, received from the Facebook servers in the JSON format. We used GET requests and stored the results within a MySQL database. We made sure to collect all the comments that were made upon each page post, which resulted in more than 1 million public comments posted by nearly 600,000 users.

Our goal was to use the content that is available within the collected dataset of 1 million comments, published on the set of 15 public pages, in order to identify spam comments via our proposed method and thus being able to build a dataset of spammer and non-spammer accounts (see Section 4). For this aim, we first decided to manually identify a number of spam comments made upon feed posts on the popular Facebook pages, and then analyze them and retrieve some attributes that are most likely able to identify spammers. Based on these attributes, we built a scoring function, described in Section 4, which enabled us to easily have a high number of suspicious candidate accounts and non-suspicious candidate accounts, on which we performed manual validation in order to build a dataset comprising, respectively, spammer and non-spammer accounts.

4. Identifying Potential Spammers and Non-Spammers Using a Scoring Method

The first step was to manually identify a set of spammers via 300 spam comments that we found directly on the 15 Facebook pages. We also chose a set of 300 non-spam comments by randomly selecting them out of the 1 million total set of comments. After that, we observed the text content along with the spammers' behavior on their walls and extracted a set of features that are likely to tell spammers apart. The purpose of this step was to help us decide on some relevant features that characterize spam content. We later proposed a score for each feature depending on how common the feature was among the identified spammers. The higher the score, the more relevant the feature was. Other commonly used features from literature were also considered as potentially relevant, as they manifested in most of the spam comments that we studied.

4.1. Features for the Scoring Method

The features that we used within the scoring function were checked for each posted comment. The scores that are given to our features vary depending on the relevance of each feature and are expressed using the integer parameter p_0. During experimentation, we set p_0 to the chosen value of 5. The following is the final list of the features:

- Overusing uppercase letters: we noticed that a lot of spammers tend to overuse uppercase letters, probably to draw the attention of other users (Figure 1). A score $S = 4 \times p_0$ is attributed to the posted comment if more than 50% of the words in it contain at least one uppercase letter.
- Commenting in another language: since we were only working with Facebook pages that publish content in English, we assumed that any user who comments in another language has good chances of being a spammer. We used the Pear open source library *Text_LanguageDetect* [34] as a language detector. Because language detectors do not always provide accurate results, we considered that a comment was not written in English if it didn't appear as a result within the top 3 most probable languages that are returned, in which case we attribute a score $S = 10 \times p_0$ to the comment. As shown in Figure 2, a high percentage of the spam-content is written in a language other than English.
- Using URLs: a score S is attributed if at least one URL is detected within the comment, such as $S = \mathit{number\_of\_URLs} \times p_0$. A higher score $S = 20 \times p_0$ might be attributed if there is no text accompanying the URL within the comment. As we may notice in Figure 3, all the 300 spam comments that were used contain URLs and the majority of them don't add any text next to the URL. On the other hand, most non-spam comments don't use URLs.
- Containing special characters and symbols: a score is attributed to the comment according to the number of special characters and symbols it contains, such as the score $S = \mathit{number\_of\_chars}$, as we noticed that a lot of spam messages contained them. Figure 4 shows that higher scores were given to spam data regarding the use of special characters compared with non-spam data.
- Containing an email address: an important score $S = 20 \times p_0$ is attributed to the comment if any email address is found.
- Containing a blacklisted word: out of the 100 observed spam comments, we retrieved frequently used words by spam-

mers such as "visit", "free", "click", *etc.* A score is attributed to the comment depending on the number of found blacklisted words, such as S = *number_of_blacklisted_words* × 5 × p_0.

- Containing a blacklisted expression: similarly to the blacklisted words, some expressions were also blacklisted, such as: "check this", "try this", "help us", "get unlimited". The attributed score also depends on the number of found expressions, therefore S = *number_of_blacklisted_expressions* × 20 × p_0. Figure 5 shows the scores attributed to spammers and non-spammers regarding the use of blacklisted words (previous feature) and expressions. We can clearly notice that the non-spam content rarely has such words or expressions.

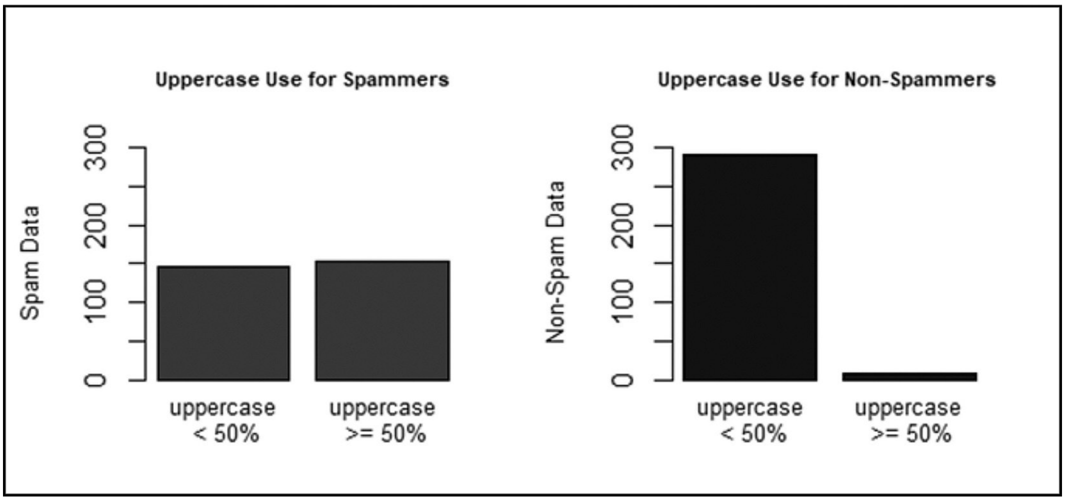


Figure 1. Uppercase use percentages for spammers and non-spammers.

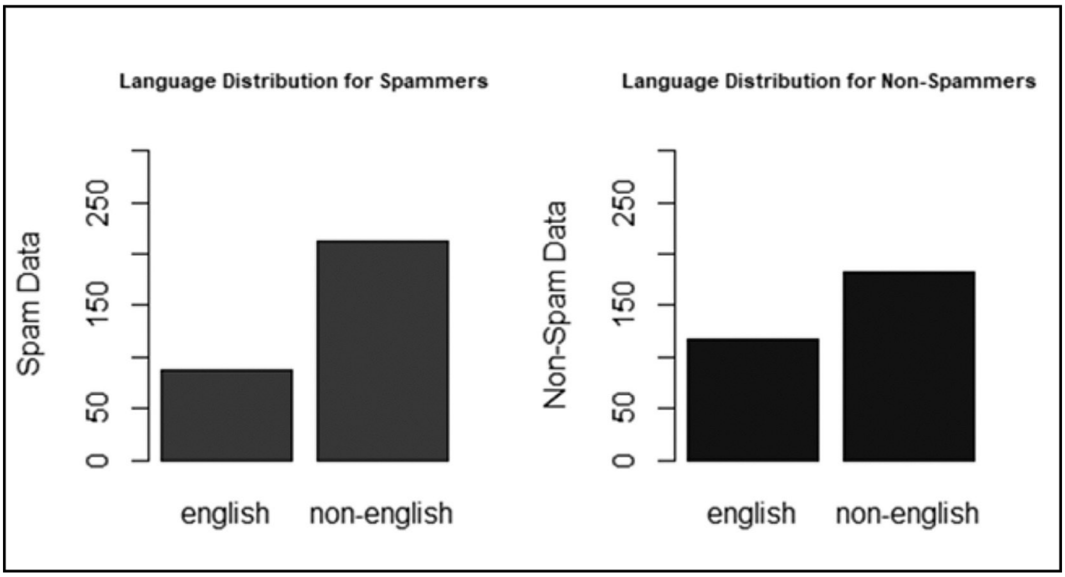


Figure 2. Language use distribution for spammers and non-spammers.

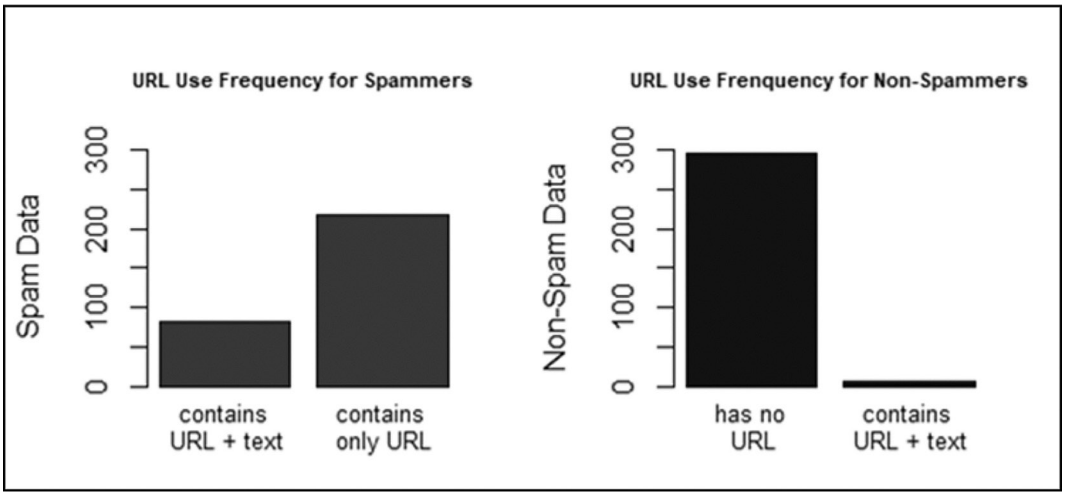


Figure 3. URL use for spammers and non-spammers.

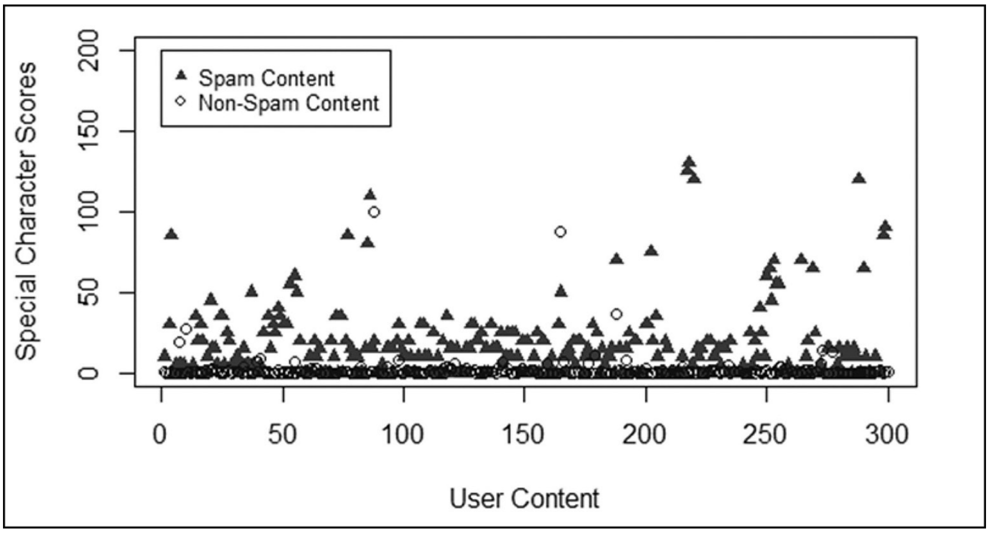


Figure 4. Special character use for spammers and non-spammers.

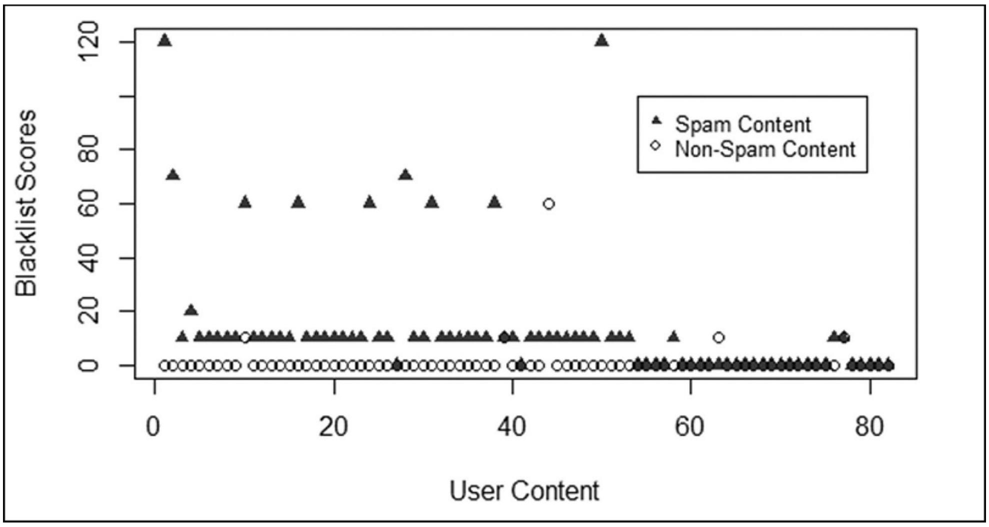
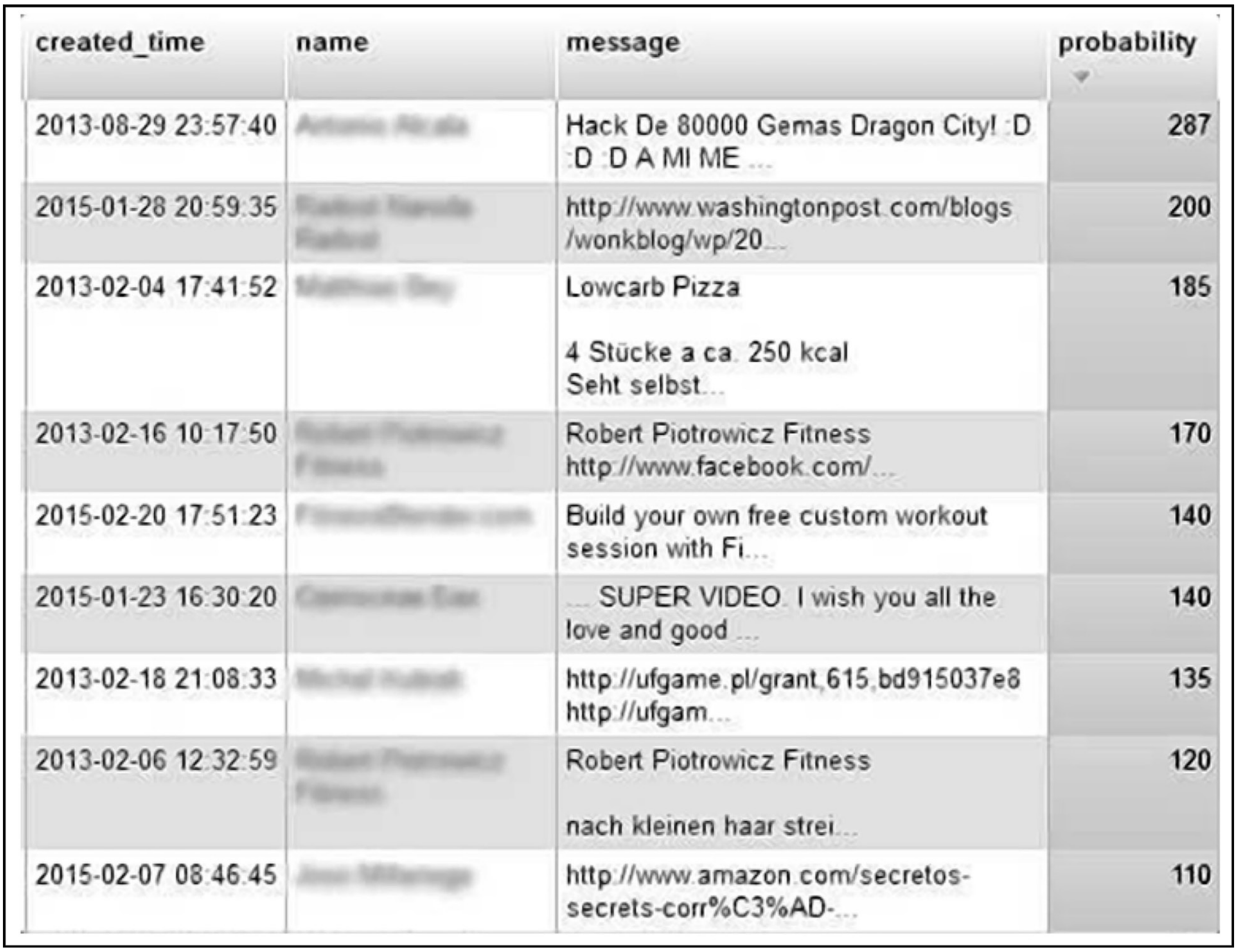


Figure 5. Blacklist words and expressions scores for spammers and non-spammers.

The scoring function was executed on the whole set of 1 million comments. For each comment, the scoring function checks each of the previous features and sums the scores generated for them at the end. The resulting score, noted in Figure 6 as "probability" (although it doesn't really represent a probability) reflects the chances of a comment being spam depending on how high its value is. After obtaining a spam likelihood score for each comment, we calculated the average score for each user, using all the scores for comments that were published by that same user, so we obtained an assessment of spam likelihood that characterizes the user itself. The user score reflects the chances of a user being a spammer: the higher its value is, the more likely a user is a spammer. We also *doubled* this score for users that have at least one comment that is duplicated. Finally, we sorted our nearly 600,000 users according to their average scores, in a descending order, from the most likely suspicious to the least likely suspicious.

4.2. Manual Validation and Final Dataset Creation

We manually validated the first 1000 users from the score-ordered list of 600,000 users that was generated by the scoring process. These top 1000 users are assumed to be most likely spammers and have the highest scores. We found that 96.2% are indeed spammers, while just 38 accounts out of 1000 were authentic. We also manually validated the last 1000 users from the bottom of the ordered 600,000 users list, which have the lowest scores and are assumed to be



created_time	name	message	probability
2013-08-29 23:57:40	[redacted]	Hack De 80000 Gemas Dragon City! :D :D :D A MI ME ...	287
2015-01-28 20:59:35	[redacted]	http://www.washingtonpost.com/blogs /wonkblog/wp/20...	200
2013-02-04 17:41:52	[redacted]	Lowcarb Pizza 4 Stücke a ca. 250 kcal Seht selbst...	185
2013-02-16 10:17:50	[redacted]	Robert Piotrowicz Fitness http://www.facebook.com/...	170
2015-02-20 17:51:23	[redacted]	Build your own free custom workout session with Fi...	140
2015-01-23 16:30:20	[redacted]	... SUPER VIDEO. I wish you all the love and good ...	140
2013-02-18 21:08:33	[redacted]	http://ufgame.pl/grant,615,bd915037e8 http://ufgam...	135
2013-02-06 12:32:59	[redacted]	Robert Piotrowicz Fitness nach kleinen haar strei...	120
2015-02-07 08:46:45	[redacted]	http://www.amazon.com/secretos- secrets-corr%C3%AD-...	110

Figure 6. A fragment of comments with spam-likelihood scores (*probability*) obtained after applying the scoring function. The comments are ordered according to *probability*, from the most likely suspicious (highest value) to the least likely suspicious (lowest value). We can notice that all the comments in the figure are indeed spam.

authentic (non-spammers). We found that 100% of those users are indeed real authentic ones. Moreover, we sampled 1000 users randomly from the set of 600,000 users and verified the users manually, then compared our manual classification (spammer vs non-spammer) with the results of the scoring function (scores) that were already given for these user samples. In Figure 7, we plotted the function's scores for the 1000 users and attributed the triangular dots to users that we manually tagged as spammers and the circular dots to users that we tagged as non-spammers.

Also, the CDF curves in Figure 8 show that spammers get generally higher scores compared to non-spammers.

The scoring function would be of a great help to quickly find potential spammer content/accounts as candidates for human verification, and thus quickly build tagged spammer datasets, which are generally hard to build, especially on Facebook.

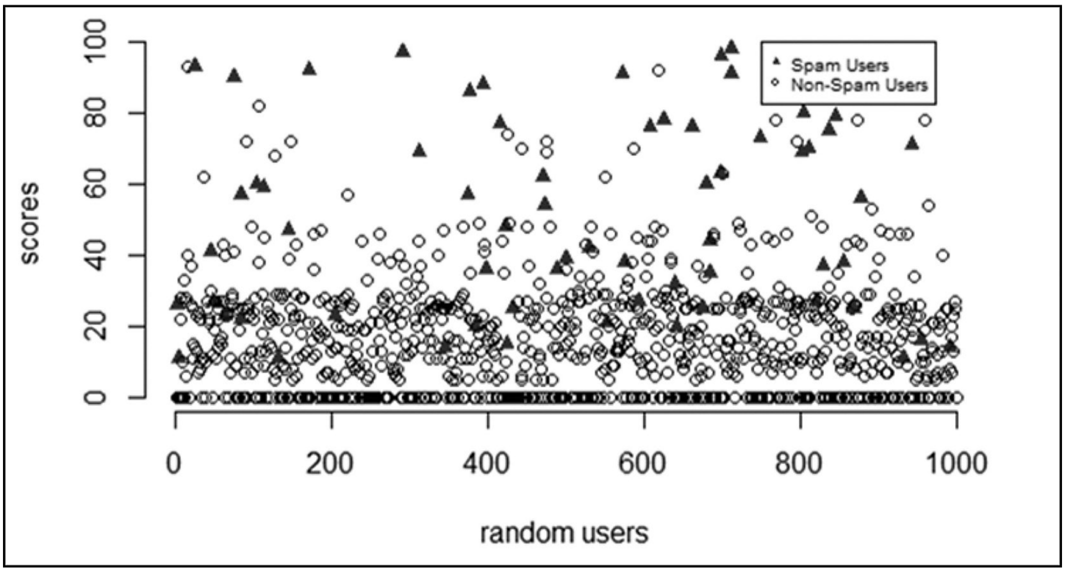


Figure 7. Scores of 1000 manually classified random users.

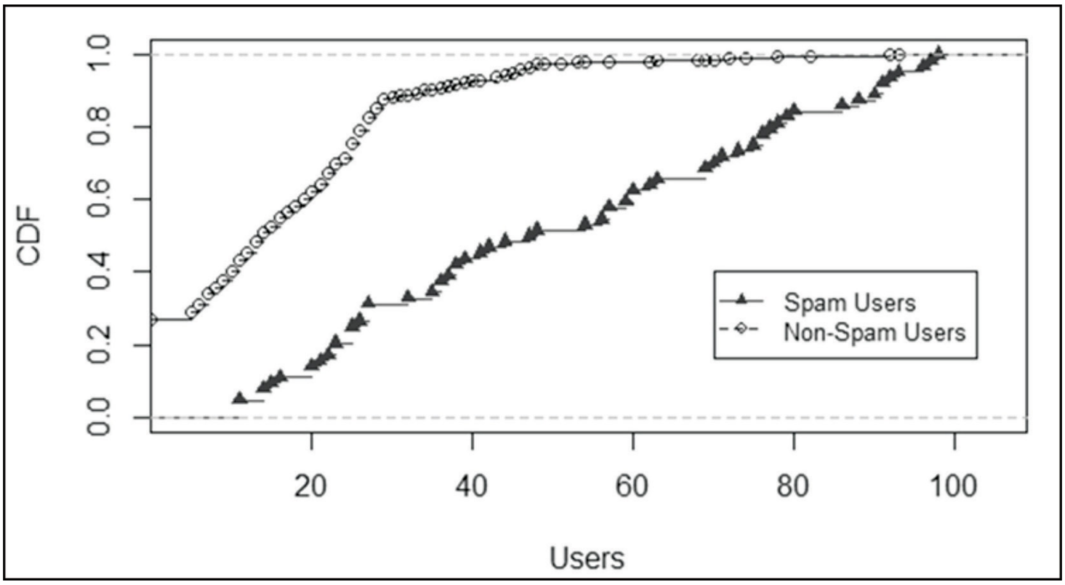


Figure 8. CDF distribution for user scores.

4.3. Brief Analysis of Spammer Accounts

We analyzed the account walls of the resulting validated sets of spam accounts and authentic accounts, and came with the following findings:

- The temporal activity patterns for spammers and authentic users are almost similar: as shown in Figure 9, we only noticed small differences between 00:00 and 04:00, where spammers tend to be a little bit more active then authentic users, and between 12:00 and 16:00 where they are slightly less active;
- There are few interactions between spammers: as shown in Figure 10, we traced a graph where vertices represent the 1000 spammers of our dataset, and each edge is traced if there was any interaction between two spammers. There are three types of interactions that we took into consideration:
  a) a spammer tagging another spammer;
  b) a spammer commenting on another spammer's wall;
  c) a spammer liking another spammer's post.

  We noticed that some edges were created, along with an absence of cycles. This might indicate that spammers in our dataset do not emulate a tightly knit community-like behavior.

5. URL-Based Identification of Groups of Spammers

In this part of the work, our aim is to detect groups of spammers who are active on Facebook pages. By "group", we mean a set of spammers that are likely controlled by one entity. Since most spammers use URLs, we assumed that those accounts that are part of a spam group most likely share similar URLs.

In our approach, we took a URL from each single user. Our idea consists of breaking down each user's URL into different *terms* and calculating the similarity between each couple of different URLs by comparing their respective sets of terms with each other. Also, we attempted incorporating in the approach the similarity between texts accompanying each URL in the comment.

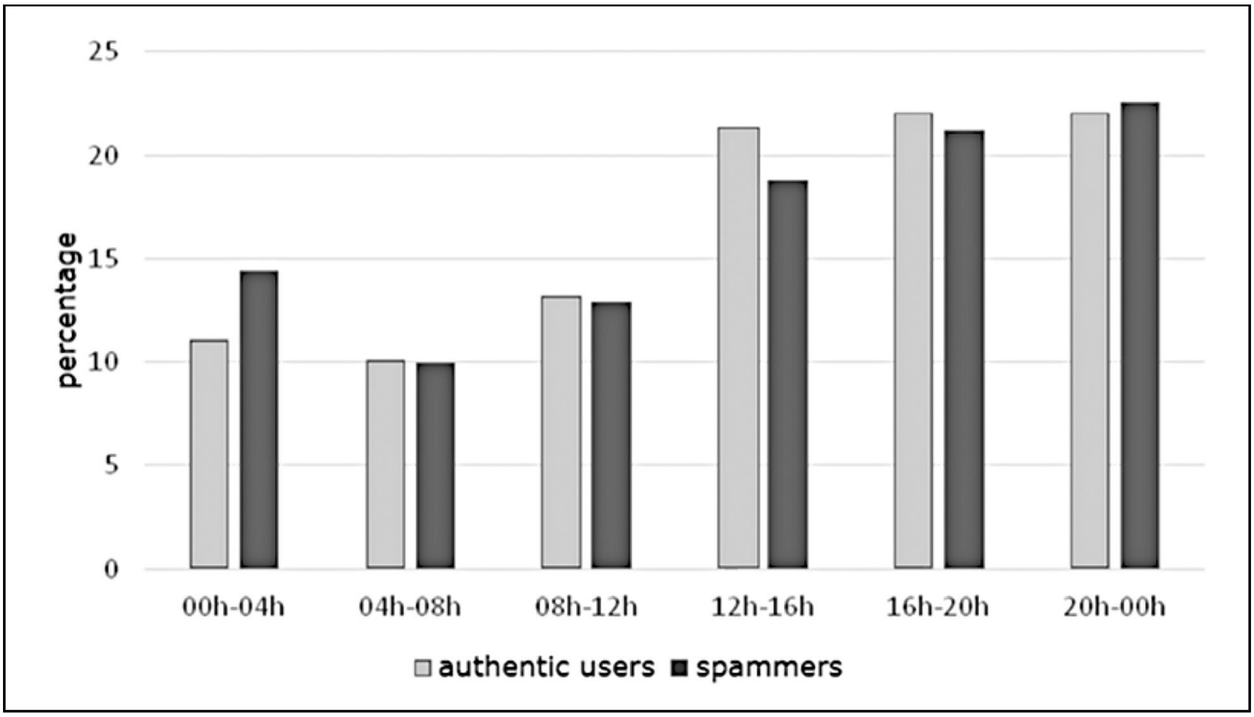


Figure 9. Percentages of published posts by authentic users and spammers, respectively, within different 4 hour segments of the day. Spammers tend to be slightly more active then authentic users between 00:00 and 04:00, and less active between 12:00 and 16:00.

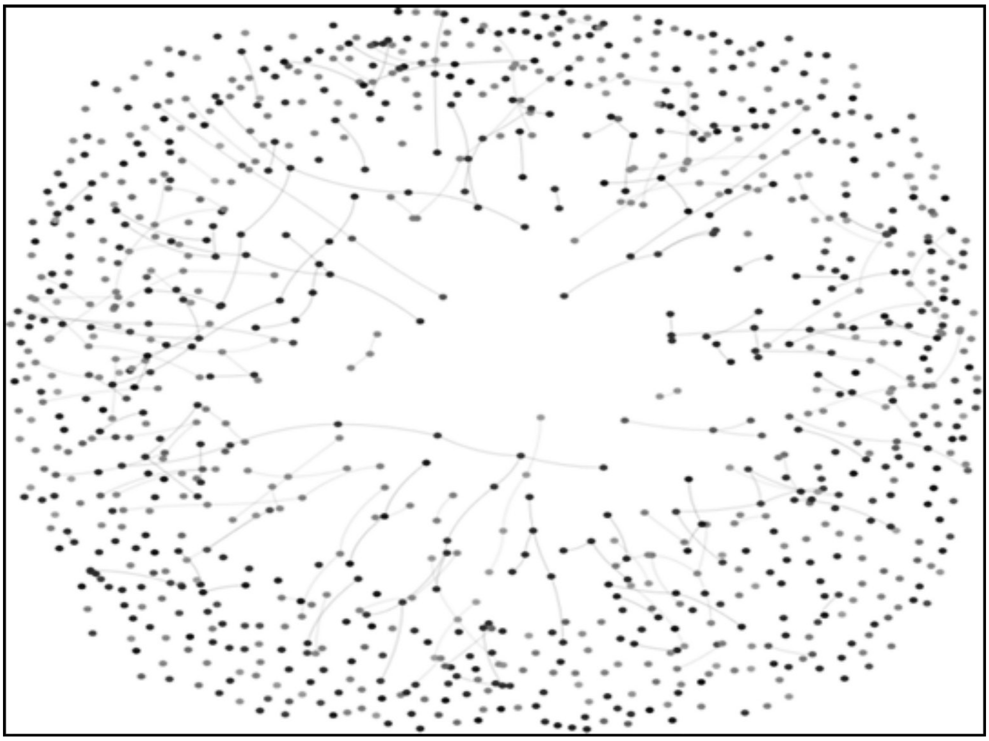


Figure 10. Social interactions between spammers. Vertices represent spammers and each edge represents an interaction between spammers through tagging, commenting or liking activities. As the figure shows, spammers do not show tight community-like interaction patterns.

5.1. Used Data

Upon investigating a few comment sections on the 15 Facebook public pages that we used in data collection, we noticed that there are some cases of multiple spammers collaborating, for instance, to promote products and websites. We also noticed that such spam content is especially present on two of the 15 public pages, both of them surpassing the bar of 1 million subscribers. For this reason, we chose to focus only on the comments that were posted on those two pages. Also, since our work will focus on URLs, we only took comments that do contain a URL. As a result, we managed to isolate a set of 9662 users. Finally, to reduce the processing cost for the next steps, we only kept 3500 users in our final dataset, which we used for applying the different approaches described within the next subsections.

5.2. Linking Users with Identical URLs

Despite developing our own approach to detect groups of spammers, we decided to reproduce the idea of linking users with completely identical URLs within a graph [35]. We used the dataset of 3500 users and plotted a graph using the tool Gephi [36], where vertices are users, characterized by their URLs, and each edge linking two users means that they share an identical URL. As we notice in Figure 11, only 3 small groups with respectively 2, 2 and 6 accounts were detected.

5.3. Our Approach: Linking Users with Similar URLs

In our approach, we are only interested in the hostname and the path of a URL, such as any given URL has generally the form: **host [/path] [?query]**. We break down the host and the path into terms if we encounter any separators among {'-','/', '.'}, and we exclude any prefixes (*e.g.*, "www"), suffixes (*e.g.*, "org"), extensions (*e.g.*, "html") and stop words (*e.g.*, "the"). For instance, for the URL: "http://subdomain.domain.com/example/product", the terms are retrieved as follows:

host: *term1* = "subdomain", *term2* = "domain",

path: *term1* = "example", *term2* = "product".

The idea of our approach is tracing a graph of users as vertices, where each edge is weighted and means that there is at least one term in common between the two users' respective URLs. Therefore, we link users not only if they share completely identical URLs, but also if they share similar URLs. After building our URL similarity graph, the problem of identifying groups of spammers reduces to a problem of clustering.

5.3.1. Algorithm

We propose our algorithm (Algorithm 1) to build our desired graph. For each couple of URLs, representing two different users, if the URLs are completely identical, then we link their respective users in the graph with an important weight $weight_0$. If not, we decompose each URL's host and path into terms, and then compare different terms to see if there are any in common. If there is at least one in common between the first URL and the second URL, it means a weighted edge should be traced between the corresponding users. Each time we find a term in common, the weight increases, depending on the type of term similarity, *i.e.*, whether the term is common between:

1. the first URL's host and the second's host, which implies using $weight_1$ to increase the edge's weight;
2. the first's host and the second's path or vice versa, which implies using $weight_2$ or
3. between the first's path and the second's path, which implies using $weight_3$.

Note that $weight_1 > weight_2 > weight_3$ because we assume that the content of the host of a URL characterizes the nature and the purpose of a webpage better than the path. Thus, when two hostnames from different URLs are similar, there might be high chances that the websites share a similar purpose, topic and/or are targeting the same audience, which is considered in our case as a sign of a potential collaborative behavior. During our experimentation, we set the

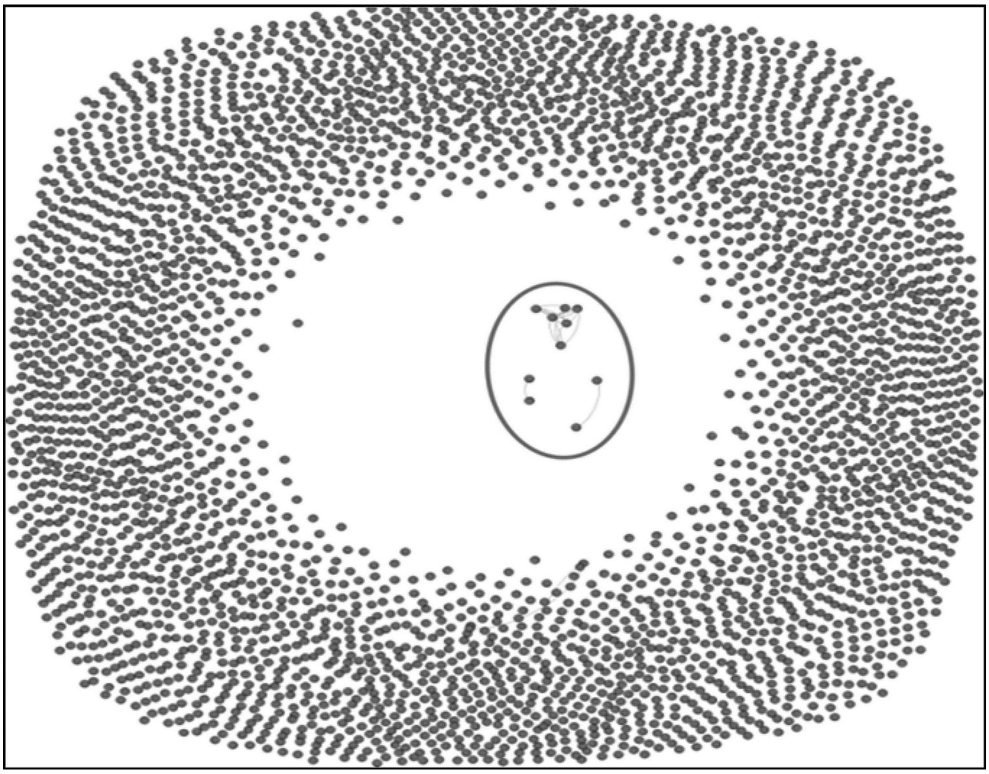
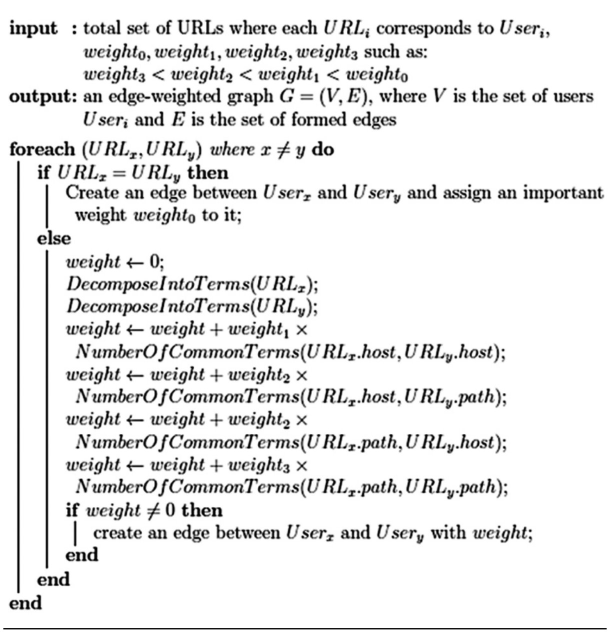


Figure 11. An illustration of users (vertices) and some rare links (edges), such that each link indicates that a couple of users shared an identical URL.

values for the weights *weight*$_0$, *weight*$_1$, *weight*$_2$, *weight*$_3$, respectively, to 1000, 100, 75 and 50.

Algorithm 1. An algorithm for building a URL similarity-based graph.



```
input  : total set of URLs where each $URL_i$ corresponds to $User_i$,
         $weight_0, weight_1, weight_2, weight_3$ such as:
         $weight_3 < weight_2 < weight_1 < weight_0$
output: an edge-weighted graph $G = (V, E)$, where $V$ is the set of users
         $User_i$ and $E$ is the set of formed edges
foreach $(URL_x, URL_y)$ where $x \neq y$ do
    if $URL_x = URL_y$ then
        Create an edge between $User_x$ and $User_y$ and assign an important
         weight $weight_0$ to it;
    else
        $weight \leftarrow 0$;
        $DecomposeIntoTerms(URL_x)$;
        $DecomposeIntoTerms(URL_y)$;
        $weight \leftarrow weight + weight_1 \times$
         $NumberOfCommonTerms(URL_x.host, URL_y.host)$;
        $weight \leftarrow weight + weight_2 \times$
         $NumberOfCommonTerms(URL_x.host, URL_y.path)$;
        $weight \leftarrow weight + weight_2 \times$
         $NumberOfCommonTerms(URL_x.path, URL_y.host)$;
        $weight \leftarrow weight + weight_3 \times$
         $NumberOfCommonTerms(URL_x.path, URL_y.path)$;
        if $weight \neq 0$ then
            create an edge between $User_x$ and $User_y$ with $weight$;
        end
    end
end
```

5.3.2. Experimentation Results

We applied our approach on the 3500 users that we kept, as stated in Subsection 5.1. We used the tool Gephi to trace the output graph for our algorithm and used the *Louvain Modularity* method it provides for clustering. We obtained the result shown in Figure 12 by changing the clustering's modularity resolution parameter until we clearly had visually distinguishable clusters, and we got 18 clusters of similar URL posters. We then isolated the clusters for further manual analysis. We were especially curious about the common terms upon which each cluster was formed. We found that among the clusters in Figure 12, the two that we surrounded by dark outlined circles with an arrow pointed to them correspond to users posting links to Facebook and YouTube. The 16 other clusters surrounded by circles were spam groups, promoting for different sites, such as the "work from home/make money online" websites, fitness products, *etc.*

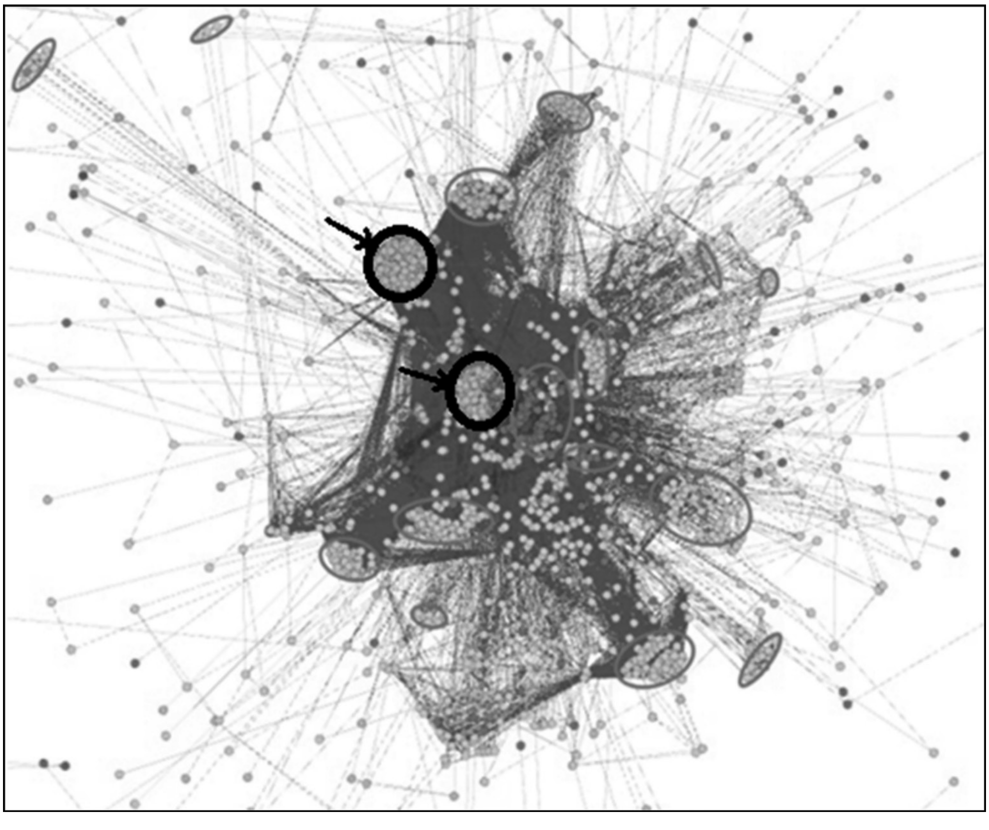


Figure 12. The identified clusters of users with our URL similarity approach. Two clusters (dark outlined circles with a pointed arrow) correspond to users posting Facebook and YouTube URLs, the other circles are spam clusters.

We also tried incorporating into our approach any similarities regarding the writing style of the textual content that is accompanying each URL within the comment. We took into account:

1. the use of upper-case letters;
2. the use of special characters, and
3. identical monetary value (*e.g.*, "make $200 from home").

However, as the results in Figure 13 show, this did not really improve results as no more clusters were found.

6. Conclusion and Future Work

In this paper, we proposed a methodology for identifying spammers among users that are active on public Facebook pages. First, since there are no available datasets of spammer Facebook accounts, we proposed a methodology relying on giving spam-likelihood scores to users who published content depending on a set of features characterizing the quality of their content and behavior. Our aim was to provide a methodology that can help researchers easily gather a set of highly likely potential spammers and non-spammers for manual validation, in order to build datasets for different purposes. Among the 1000 accounts that were selected by the methodology to be inspected for spam activity, 96.2% of the accounts were indeed tagged as spammers later, and for the 1000 non-spam set, 100% of the accounts were in fact tagged as non-spammers upon manual human inspection. Thus, that enabled us to collect a set of spammers and non-spammers, comprising each 1000 manually validated accounts, on which we performed the analysis. Our findings indicate that spammers rarely interact with each other and have almost similar temporal activity patterns to non-spammers, with slight differences.

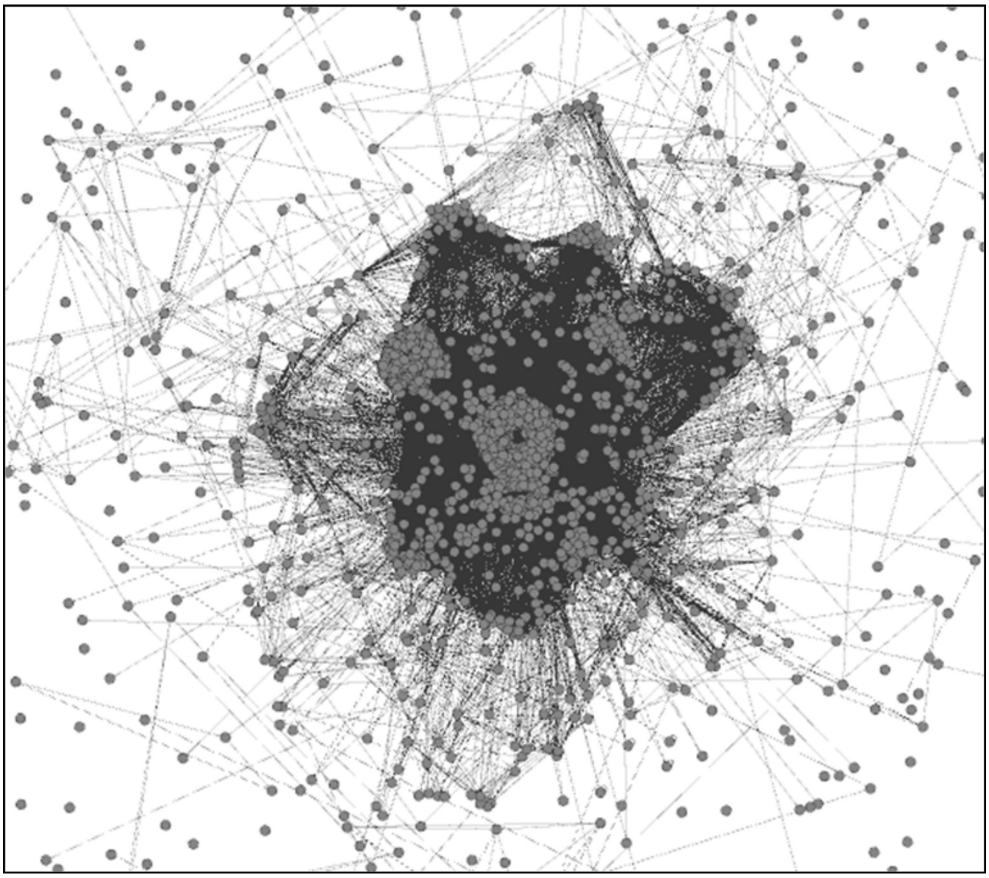


Figure 13. The results of combining the URL similarity approach with the writing style. The combination didn't improve results as no more clusters were formed.

Second, we proposed an approach for identifying clusters of spammers based on a URL similarity graph, where vertices represent users, characterized by their URLs, and weighted edges represent connected URLs that share one or more common terms, reflecting to which extent they are similar. This approach was more efficient in detecting spam groups compared to using identical URLs for edge creation, and it successfully identified 16 clusters of spammers.

In this work, public Facebook pages were assumed to make an ideal ground for spammer activity. However, we are currently interested in identifying spam pages on Facebook, where pages themselves are used to spam their subscribers and visitors. An interesting direction for future work is to use our proposed feature set in Section 4 for the detection of spam content on public pages, in order to collect a dataset of spam pages for further analysis.

References

[1] Daily Mail, "Facebook Can Take Over from Churches, Says Mark Zuckerberg", 2017. http://www.dailymail.co.uk/sciencetech/article-4644592/Facebook-2-BILLION-monthly-active-users.html

[2] USA Today, "USA Today Asks FBI to Probe Rise in Fake Facebook Followers", 2017. https://www.usatoday.com/story/tech/news/2017/05/05/usa-today-asks-fbi-probe-rise-fake-facebook-followers/101303300

[3] Adespresso, "Buying Facebook Likes Sucks, Here's the Data to Prove It!", 2016. https://adespresso.com/blog/buy-facebook-likes

[4] Darkreading, "Facebook Spam Botnet Promises Likes for Access Tokens", 2017. https://www.darkreading.com/endpoint/facebook-spam-botnet-promises-likes-for-access-tokens/d/d-id/1328756

[5] Reuters, "Facebook Says It Will Act Against Information Operations Using False Accounts", 2017. https://www.reuters.com/article/us-facebook-propaganda-response/facebook-says-it-will-act-against-information-operations-using-false-accounts-idUSKBN17T2G6

[6] T. Stein *et al.*, "Facebook Immune System", in Proc. of the 4th Workshop on Social Network Systems, pp. 8:1–8:8, 2011. https://doi.org/10.1145/1989656.1989664

[7] Facebook Help Community, "When Will Facebook Address Spam Reporting on Political Pages?", 2013. https://fr-fr.facebook.com/business/help/community/question/?id=10201714337112983

[8] Reuters, "Facebook Cracks Down on 30,000 Fake Accounts in France", 2017. http://www.reuters.com/article/france-security-facebook/facebook-cracks-down-on-30000-fake-accounts-in-france-idUSL8N1HL4IH

[9] Reuters, "Facebook Changes Algorithm to Curb Tiny Group of Spammers", 2017. https://www.reuters.com/article/us-facebook-spam/facebook-changes-algorithm-to-curb-tiny-group-of-spammers-idUSKBN19L2LA

[10] The Guardian, "How Facebook Powers Money Machines for Obscure Political News Sites", 2016. https://www.theguardian.com/technology/2016/aug/24/facebook-clickbait-political-news-sites-us-election-trump

[11] F. Ahmed and M. Abulaish, "A Generic Statistical Approach for Spam Detection in Online Social Networks", *Computer Communications*, vol. 36, no. 10-11, pp. 1120–1129, 2013. https://doi.org/10.1016/j.comcom.2013.04.004

[12] A. Kumbhar *et al.*, "A Survey on: Malicious Application and Fake User Detection in Facebook Using Data Mining", *International Journal of Engineering Science and Computing*, vol. 7, pp. 15768–15771, 2017.

[13] K. Lee *et al.*, "Uncovering Social Spammers: Social Honeypots + Machine Learning", in *Proc. of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, 2010, pp. 435-442. https://doi.org/10.1145/1835449.1835522

[14] F. Brito *et al.*, "Detecting Social-Network Bots Based on Multiscale Behavioral Analysis", in *Proc. of SECURWARE 2013: The Seventh International Conference on Emerging Security Information, Systems and Technologies*, 2013, pp. 81–85.

[15] Z. Chu *et al.*, "Who is Tweeting on Twitter: Human, Bot, or Cyborg?", in *Proc. of the 26th Annual Computer Security Applications Conference*, 2010, pp. 21–30. https://doi.org/10.1145/1920261.1920265

[16] A. H. Wang, "Detecting Spam Bots in Online Social Networking Sites: a Machine Learning Approach", *Data and Applications Security and Privacy XXIV*, vol. 6166 LNCS, pp. 335–342, 2010. https://doi.org/10.1007/978-3-642-13739-6_25

[17] Z. Yang *et al.*, "Uncovering Social Network Sybils in the Wild", in *Proc. of the 2011 ACM SIGCOMM conference on Internet measurement*, 2011, pp. 259–268. https://doi.org/10.1145/2556609

[18] F. Benevenuto *et al.*, "Detecting Spammers and Content Promoters in Online Video Social Networks", in *Proc. of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, 2009, pp. 620–627. https://doi.org/10.1109/INFCOMW.2009.5072127

[19] M. Mohammadrezaei *et al.*, "Identifying Fake Accounts on Social Networks Based on Graph Analysis and Classification Algorithms", *Security and Communication Networks*, vol. 2018, pp. 5923156:1–5923156:8, 2018. https://doi.org/10.1155/2018/5923156

[20] Y. Boshmaf *et al.*, "The socialbot Network: When Bots Socialize for Fame and Money", in *Proc. of the 27th Annual Computer Security Applications Conference*, 2011, pp. 93–102. https://doi.org/10.1145/2076732.2076746

[21] H. Yu *et al.*, "SybilGuard: Defending Against Sybil Attacks via Social Networks", *IEEE/ACM Transactions on Networking*, vol. 16, pp. 576–589, 2008. https://doi.org/10.1109/TNET.2008.923723

[22] B. Wang *et al.*, "SybilSCAR: Sybil Detection in Online Social Networks via Local Rule Based Propagation", in *Proc. of the IEEE International Conference on Computer Communications INFOCOM 2017*, 2017, pp. 1–9. https://doi.org/10.1109/INFOCOM.2017.8057066

[23] H. Yu *et al.*, "SybilLimit: A Near-Optimal Social Network Defense Against Sybil Attacks", *IEEE/ACM Transactions on Networking*, vol. 18, 2010, pp. 885–898. https://doi.org/10.1109/SP.2008.13

[24] N. Z. Gong *et al.*, "SybilBelief: A Semi-Supervised Learning Approach for Structure-Based Sybil Detection", *IEEE Transactions on Information Forensics and Security*, vol 9, 2014, pp. 976–987. https://doi.org/10.1109/TIFS.2014.2316975

[25] A. Beutel *et al.*, "Copy-Catch: Stopping Group Attacks by Spotting Lockstep Behavior in Social Networks", in *Proc. of the 22nd World Wide Web Conference*, 2013, pp. 119–130. https://doi.org/10.1145/2488388.2488400

[26] Q. Cao *et al.*, "Uncovering Large Groups of Active Malicious Accounts in Online Social Networks", in *Proc. of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, 2014, pp. 477–488. https://doi.org/10.1145/2660267.2660269

[27] G. Wang *et al.*, "You Are How You Click: Clickstream Analysis for Sybil Detection", in *Proc. of the 22nd USENIX conference on Security*, 2013, pp. 241–256.

[28] F. Ahmed and M. Abulaish, "Identification of Sybil Communities Generating Context-Aware Spam on Online Social Networks", in *Ishikawa Y., Li J., Wang W., Zhang R., Zhang W. (Eds): Web Technologies and Applications, APWeb 2013, Lecture Notes in Computer Science*, vol. 7808, Springer, Berlin, Heidelberg, 2013, pp. 268–279. https://doi.org/10.1007/978-3-642-37401-2_28

[29] H. Gao *et al.*, "Detecting and Characterizing Social Spam Campaigns", in *Proc. of the 10th ACM SIGCOMM conference on Internet measurement*, 2010, pp. 35–47. https://doi.org/10.1145/1879141.1879147

[30] L. Fulu *et al.*, "The Community Behavior of Spammers", 2011. http://web.media.mit.edu/~fulu/ClusteringSpammers.pdf

[31] CBS News, "Facebook Purges Thousands of Fake Accounts that Targeted Publishers", 2017. https://www.cbsnews.com/news/facebook-purges-fake-accounts-publishers

[32] New Scientist, "Data Sifted From Facebook Wiped After Legal Threats", 2010. https://www.newscientist.com/article/dn18721-data-sifted-from-facebook-wiped-after-legal-threats

[33] Facebook for Developers. https://developers.facebook.com

[34] PEAR-PHP. https://pear.php.net/package/Text_LanguageDetect

[35] F. Li and M. Hsieh, "An Empirical Study of Clustering Behavior of Spammers and Groupbased anti-spam Strategies", in *Proc. of CEAS 2006 Third Conference on Email and AntiSpam*, 2006, pp. 27–28.

[36] Gephi. https://gephi.org

Received: March 2021
Revised: March 2022
Accepted: July 2022

Contact addresses:
Hakim Azri
Université des Sciences et de la Technologie
d'Oran - Mohamed Boudiaf
Department of Computer Science
LSSD Laboratory
Algeria
e-mail: hakim.azri@univ-usto.dz

Hafida Belbachir
Université des Sciences et de la Technologie
d'Oran - Mohamed Boudiaf
Department of Computer Science
LSSD Laboratory
Algeria
e-mail: h_belbach@yahoo.fr

Fatiha Guerroudji Meddah
Université des Sciences et de la Technologie
d'Oran - Mohamed Boudiaf
Department of Computer Science
Algeria
e-mail: fatiha.guerroudji@univ-usto.dz

Hakim Azri received his MSc degree in computer science in 2011. He is currently a PhD student at the computer science department of the Université des Sciences et de la Technologie d'Oran - Mohamed Boudiaf (USTO-MB) and a member of the LSSD laboratory. His research interests include data science, machine learning and social network analysis.

Hafida Belbachir received her PhD degree in computer science from the University of Es-Senia, Oran, Algeria, in 1990. From 1992 to 2006, she was an associate professor at the Université des Sciences et de la Technologie d'Oran – Mohamed Boudiaf (USTO-MB), Oran, Algeria. Since 2006, she is a professor at the same university. Prof. Hafida Belbachir has been the head of the Database group in the LSSD laboratory at the same university since 2007. Her research interests include advanced databases, data mining and data grid.

Fatiha Guerroudji Meddah is an associate professor at the University of Science and Technology Mohammed Boudiaf Oran (USTOMB). She received her MSc and PhD degrees in computer science from USTOMB University (Oran, Algeria). Her research interests lie in the areas of artificial intelligence, geographic information sciences, interactive cartography and geovisualization. She has several participations in projects in these areas. She has published in international journals and conferences and has been a program committee member of national and international conferences.