Combating Fraud in Online Social Networks: Characterizing and Detecting Facebook Like Farms

Muhammad Ikram[1†], Lucky Onwuzurike[2†], Shehroze Farooqi[3†], Emiliano De Cristofaro[2], Arik Friedman[1], Guillaume Jourjon[1], Mohamed Ali Kaafar[1], M. Zubair Shafiq[3]

[1] NICTA, [2] University College London, [3] University of Iowa

ABSTRACT

As businesses increasingly rely on social networking sites to engage with their customers, it is crucial to understand and counter reputation manipulation activities, including fraudulently boosting the number of Facebook page likes using so-called *like farms*. Thus, social network operators have started to deploy various fraud detection algorithms such as graph clustering methods, however, with limited efficacy. In fact, this paper presents a comprehensive analysis and evaluation of existing graph-based fraud detection algorithms for detecting like farm accounts. Our results show that more sophisticated and stealthy farms can successfully evade detection by spreading likes over longer timespans and by liking many popular pages to mimic normal users.

Next, we analyze a wide range of features extracted from users' timeline posts, which we group into two main classes: lexical and interaction-based. We find that like farm accounts tend to more often re-share content, use fewer words and poorer vocabulary, target fewer topics, and generate more (often duplicate) comments and likes compared to normal users. Using these timeline-based features, we experiment with machine learning algorithms to detect like farms accounts, obtaining appreciably high accuracy (as high as 99% precision and 97% recall).

1. INTRODUCTION

Online social networks enable organizations and individuals to reach out to large audiences, offering a number of tools to easily engage users worldwide. Among these, Facebook *pages* have emerged as an important asset for companies, businesses, and public figures, allowing them to broadcast updates, promote products/events, and get in touch with customers and fans. The number of "likes" of a Facebook page is often considered a measure of its popularity and profitability [6]. Facebook allows page owners to *promote* their pages via paid advertising campaigns: a recent report [22] claims that more than 40 million small businesses have active pages and almost 2 million of them use Facebook's advertising platform to promote them.

At the same time, a large number of so-called *"like farms"* offer paid services that artificially inflate the number of likes on a given page, often relying on a network of fake and compromised accounts [28]. Facebook estimates that 5-7% of their accounts are potentially fake [9] and also considers like farms as fraud, routinely launching clean-up campaigns to remove such likes as well as the accounts involved. In a recent study, [8] use honeypots to analyze fake likes activities and found that, while some farms follow a *naïve* approach with a large number of accounts liking target pages within a short timespan, some take a stealthy approach to gradually spread likes over longer timespans, possibly aiming to evade fraud detection algorithms identifying lockstep behavior. Authors also showed that Facebook removed only a handful of likes generated by the farms and terminated a very limited number of accounts used by them.

In this paper, we characterize the liking patterns of accounts associated with like farms and systematically evaluate the effectiveness of graph co-clustering fraud detection algorithms [1, 5] in detecting like farm accounts. Our evaluation shows that stealthy like farms can successfully circumvent these algorithms, with significantly high false positives rate. We find that spreading likes over longer timespans and liking popular pages to mimic normal users make it quite difficult to detect like farm activities by only relying on their liking patterns.

Consequently, we set to investigate whether additional timeline information, including lexical and interaction characteristics of timeline posts, can help in accurately detection like farm accounts. To this end, we crawl and analyze timelines of user accounts associated with like farms as well as a baseline of normal user accounts. We find that posts made by like farm accounts have fewer words, a more limited vocabulary, and lower readability than normal users' posts. Moreover, like farm posts are highly targeted to some specific topics, generate significantly more comments and likes, and a large fraction of their posts consists of non original and often redundant "shared activity" (i.e., repeatedly sharing posts made by other users, articles, videos, and external URLs).

Based on these original timeline-based features, we build three classifiers using supervised two-class support vector machines (SVM) [17] and evaluate them using our ground-truth dataset. Our first and second classifiers are fed, respectively, with lexical and interaction features extracted from

[†] Authors contributed equally.

timeline posts, while the third one uses both. Our evaluation shows that the latter can accurately detect like farms accounts, achieving 99% average precision and 97% recall.

2. DATA

Campaigns. Our starting point is the data collected by [8]: they created 13 honeypot Facebook pages called *"Virtual Electricity"* and, while keeping them empty (i.e., no posts/pictures), they promoted them using legitimate Facebook page like ads (5 Facebook pages out of 13) and popular like farms (8 out of 13). More specifically, five Facebook campaigns targeted, respectively, users in U.S, France, India, Egypt, and worldwide, while the other eight employed BoostLikes.com (BL), SocialFormula.com (SF), AuthenticLikes.com (AL), and MammothSocials.com (MS) farms, with one campaign per farm targeting worldwide and one U.S users. In the rest of the paper, we use the campaign acronyms followed by their target, e.g., the label SF-ALL denotes the SocialFormula campaign targeting worldwide users. BL-ALL and MS-ALL did not actually deliver any likes, even though they were paid.

Overall, the pages attracted likes from a total of 5,918 users out of which 5,616 are unique accounts (1,437 from Facebook ads and 4,179 from the like farm campaigns) as some users liked more than one of the honeypot pages. Note that, out of the 5,616 accounts, we found that 642 (≈11%) accounts had become inactive by August 2015. Besides the accounts attracted by the honeypots, our analysis also uses a sample of 1,408 random accounts collected by [7], which form a baseline of "normal" accounts.

Likes. For each of the accounts liking a honeypot page, [8] also crawled, using Selenium web driver[1], the other pages that the account liked. In order to get an updated corpus, *between August and October 2015, we have re-crawled* the pages liked for each of the farm users and, using the page identifier information, we also collect the information associated with each page – i.e., total number of likes, category and location. We do the same for the baseline accounts ([7]). Overall, we gather information from more than 1.1 million pages.

Timelines. Whenever it is publicly available, we also crawl the timeline information (aka *Facebook wall*) of each user—which was not done by [8]. In particular, we crawl timeline posts (capping at a maximum of 500 posts), the latest comments on each post, as well as the associated number of likes it received and the number of comments. In total, we collect more than 336K posts (messages, shared content, check-ins, etc.) from like farm accounts, and more than 35K posts from the baseline accounts.

Summary. Table 1 summarizes the accounts we use for our analysis. Note that it reflects a double-count of users that like more than one of the honeypot pages. In the remainder of the paper, we will use the campaign origin from Table 1 as the label to train and evaluate our various algorithms. In the case of accounts belonging to multiple campaigns, these accounts will be replicated across these campaigns.

[1] http://docs.seleniumhq.org/projects/webdriver/

Campaign	#Users	#Pages Liked	#Unique Pages Liked	#Posts
BL-USA	583	79,025	37,283	44,566
SF-ALL	870	879,369	108,020	46,394
SF-USA	653	340, 964	75,404	38,999
AL-ALL	707	162,686	46,230	61,575
AL-USA	827	441,187	141,214	30,715
MS-USA	259	412,258	141,262	12,280
Baseline	1,408	79,247	57,384	34,903

Table 1: Overview of the datasets used in our study.

Ethical Considerations. We remark that we only collected openly available data such as (public) profile and timeline information, as well as page likes. We also enforced a few mechanisms to protect user privacy: all data was encrypted at rest and not re-distributed, and no personal information was extracted as we only analyzed aggregated statistics. We also requested clearance from the local Institutional Review Board (IRB), which classified our research as exempt.

3. BACKGROUND & LIMITATIONS OF PRIOR WORK

Aiming to mitigate issues like page like fraud, Facebook has rolled out detection tools such as CopyCatch [1] and SynchroTrap [5]. The two are graph co-clustering algorithms geared to detect large groups of malicious accounts that like similar pages around the same time frame.

As highlighted by [8], however, some like farm operators have modified their behavior to avoid synchronized patterns. Specifically, while several farms follow a *naïve* approach with a large number of accounts (possibly fake or compromised) liking target pages within a short timespan, some take a *stealthier* approach spreading likes over longer timespans and liking popular pages to circumvent fraud detection algorithms.

3.1 Graph Co-Clustering Technique

We first attempt to evaluate the effectiveness of user-page graph co-clustering algorithms, like those deployed by Facebook to detect fake likes by malicious users. We use the labeled dataset of 4,179 users from six different like farms and the 1,408 baseline users (see previous section). We employ a graph co-clustering algorithm to divide the user-page bipartite graph into distinct clusters [15]. Similar to CopyCatch [1] and SynchroTrap [5], the identified clusters in the user-page bipartite graph represent near-bipartite cores, and the set of users in a near-bipartite core like the same set of pages. Since we are interested in distinguishing between two classes of users (like farm users and normal users), we set the target number of clusters at 2.

In Table 2, we report the ROC statistics (TP: true positive, FP, false positive, TN: true negative, FN: false negative, Pre-

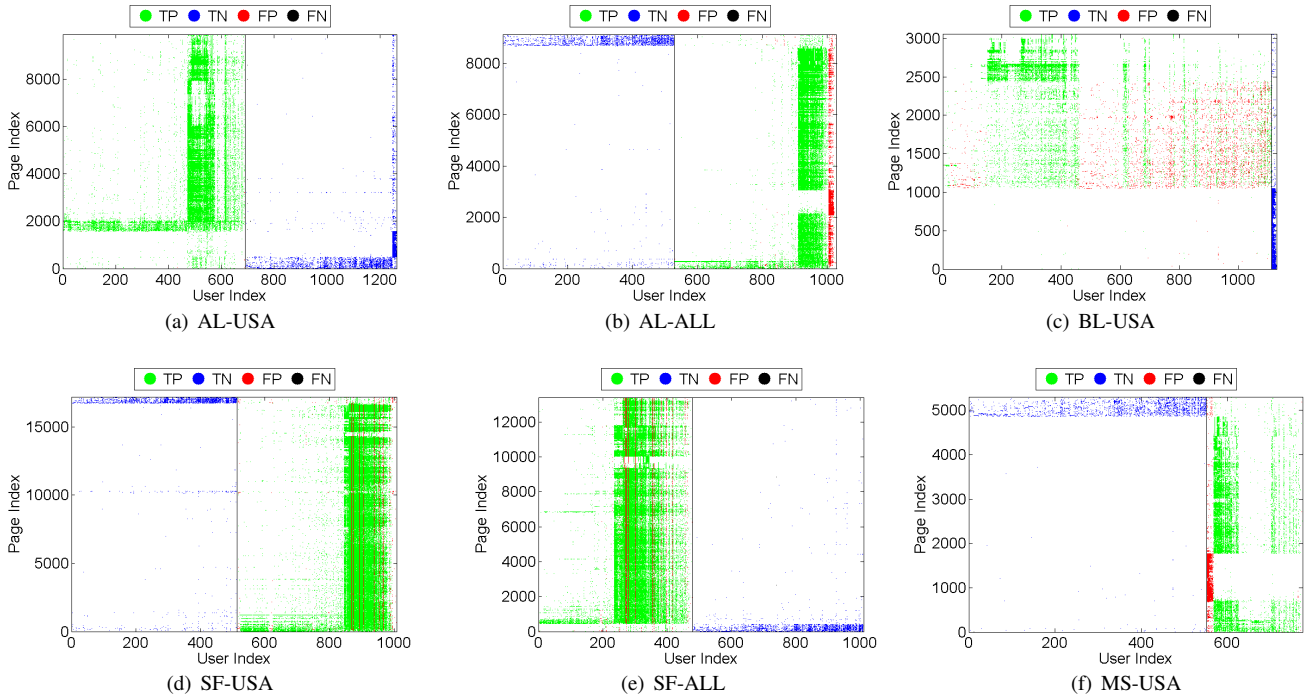


Figure 1: Visualization of graph co-clustering results. The vertical black line indicates the separation between two clusters. We note that the clustering algorithm fails to achieve good separation leading to a large number of false positives (red dots).

Campaign	TP	FP	TN	FN	Precision	Recall	F1-measure
AL-USA	681	9	569	4	98%	99%	99%
AL-ALL	448	53	527	1	89%	99%	94%
BL-USA	523	588	18	0	**47%**	100%	64%
SF-USA	428	67	512	1	86%	100%	94%
SF-ALL	431	48	530	2	90%	99%	95%
MS-USA	201	22	549	2	90%	99%	93%

Table 2: Effectiveness of the graph co-clustering algorithm.

cision: $\frac{TP}{TP+FP}$, Recall: $\frac{TP}{TP+TN}$, F1-measure: harmonic average of precision and recall) of the graph co-clustering algorithm. Figure 1 visualizes the clustering results as user-page scatter plots. The x-axis represents user index and the y-axis represents page index.[2] The vertical black line marks the separation between two clusters. The points in the scatter plot are colored to indicate true positives (green), true negatives (blue), false positives (red), and false negatives (black).

We observe two distinct behaviors in the scatter plots: (1) *"liking everything"* (vertical streaks), and (2) *"everyone liking a particular page"* (horizontal streaks). Both like farm and normal users exhibit vertical and horizontal streaks in the scatter plots. While the graph co-clustering algorithm neatly separates users for AL-USA, it incurs false positives for other like farms. In particular, the co-clustering algorithm fails to achieve a good separation for BL-USA, where it incurs a large number of false positives, resulting in 47% precision. Further analysis reveals that the horizontal false positive streaks in BL-USA include popular pages, such as "Fast & Furious" and "SpongeBob SquarePants," each with millions of likes. We deduce that stealthy like farms such as BL-USA use the tactic of liking popular pages to mimic normal users, which confuses the graph co-clustering algorithm.

Next, we set to validate whether text-based features of timelines posts can improve the accuracy of detection. We do so by using a latent semantic analysis approach, "bag-of-words" analysis, and use a machine learning classifier to distinguish normal users (i.e. genuine "likers" of pages) from campaign users.

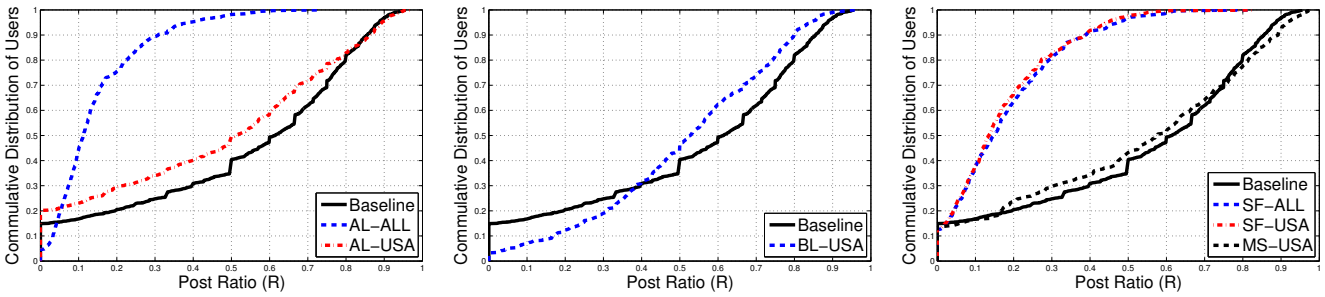
3.2 Word Frequency Technique

In the following we consider user timelines as the collection of posts and the corresponding comments on each post we collected (i.e. all textual content of timelines) . We build a corpus of words extracted from user timelines by applying the term frequency-inverse document frequency (TF-IDF) statistical tool [18]. To build both testing and training set, we consider user timeline as unique documents belonging to either like farm users or normal users, then, we apply TF-IDF after removing all English stop-words. We form two classes by labeling like farm and baseline users' TF-IDF features as positives and negatives, respectively, and use respectively, 80% and 20% of their TF-IDF features to create training and testing sets. Using the TF-IDF features, we train

[2]To ease presentation, we exclude users and pages with less than 10 likes.

Campaign	Total Users	Training Set	Testing Set	TP	FP	TN	FN	Precision	Recall	Accuracy	F1-Measure
AL-USA	827	661	204	103	9	229	101	92%	50%	75%	65%
AL-ALL	707	566	141	101	1	237	40	99%	72%	89%	83%
BL-USA	583	468	115	78	1	237	37	99%	68%	89%	80%
SF-USA	652	522	130	83	0	238	47	100%	89%	73%	84%
SF-ALL	870	697	173	128	3	235	45	88%	98%	74%	84%
MS-USA	259	210	49	32	5	233	17	86%	65%	92%	74%

Table 3: Effectiveness of TF-IDF based text features with SVM in detecting like farm accounts.



(a) AL (b) BL (c) MS and SF

Figure 2: Distributions of the ratio of English to non-English posts.

a two-class support vector machine (SVM) [17], relying on *scikit-learn* [3], an open source machine learning library for Python.

Table 3 shows the results of our classifier. Compared to the co-clustering algorithm, we observe that the TF-IDF based classifier improves F1-measure by approximately 16%, but only for the BL-USA campaign. This result improvement discrepancy is mainly due to poor results of the co-clustering algorithm, where the BL-USA farm have an harmonic average of precision and recall of only 64% as compared to above 90% for the other campaigns. Nonetheless, the overall performance of the "bag-of-words" approach is poor, which can be explained with the short nature of the posts. Indeed, [12] recently demonstrated that the word frequency approach to analyze short text on Twitter and on blogs does not perform well to to analyze, e.g., user sentiment.

3.3 Takeaway

Our results highlight the limitations of prior graph co-clustering algorithms in detecting fake likes by like farm accounts. We also show that a simple text-based machine learning classifier fails to improve the detection accuracy. We argue that fake likes activity is challenging to detect solely based on monitoring the liking activity and simple timeline-based text features, due to the increased sophistication of like farms. Therefore, as we discuss next, we will look at auxiliary information, such as the characteristics of timeline posts, in order to more accurately detect like farm accounts.

4. CHARACTERIZING TIMELINE FEATURES

We now analyze users' timelines and derive two categories of features: *interaction with posts* and *lexical*. We do so aiming to identify the most distinguishing features that can be used by machine learning algorithms for accurately classifying like farms and normal users.

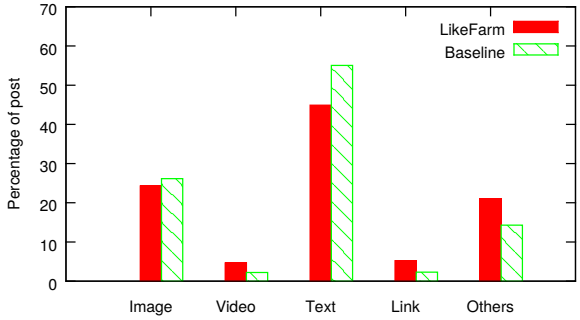


Figure 3: Distribution of types of posts.

4.1 Interactions with Posts

We analyze how users interact with the posts on their timeline with the goal of identifying distinctive timeline-based features. Interaction-based features are interesting from different perspective. First, these are features that are not dependant on the language used by the account owner, which makes their use generic enough. Second, interaction-based features do also capture a form of user behavior when involved in online social networks activity. Regardless of whether users are connecting to the network to genuinely connect with their friends or for different other purposes we are expecting to observe discrepancies between regular users and accounts involved in fraud activities such as Farm likes.

Types of Posts. Figure 3 shows the percentage of the *types* of posts that appear on users' timelines. More than 50% of posts made by baseline users are text, whereas, for like farm

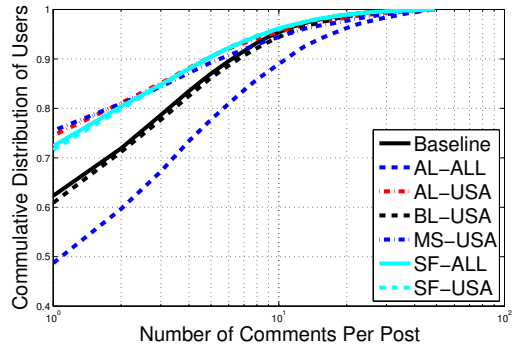


Figure 4: Distribution of number of comments per post.

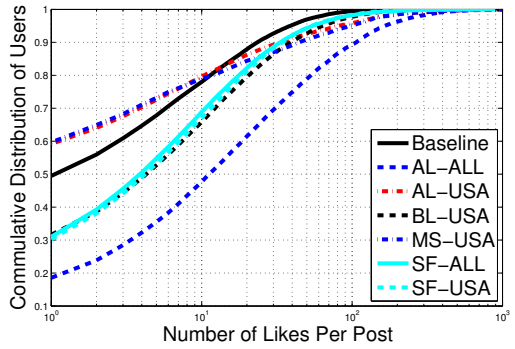


Figure 5: Distribution of number of likes per post.

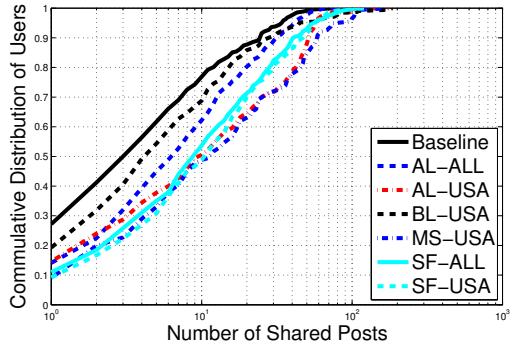


Figure 6: Distribution of number of shared posts per user.

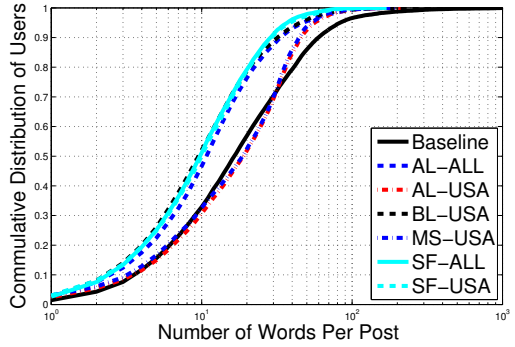


Figure 7: Distribution of number of words per text post.

users this ratio is less than 44% as they post more web links and videos. Note that "Others" include shared posts, Facebook actions such as 'listening to', 'traveling to', 'feeling' etc., and life events like 'in a relationship', and 'married'. We find that this category includes about 22% of posts for like farm users and about 16% of posts for baseline users.

Comments and Likes. Next, in Figure 4, we plot the distributions of the number of comments a post attracts, revealing that users of two like farms, i.e., AL-ALL and BL-USA, generate more comments than the baseline users with BL-USA almost identical to the baseline users indicating stealthiness. Figure 5, showing the number of likes associated with users' posts, highlights that posts of like farm users attract much more likes than those of baseline users. Therefore, posts produced by like farm users gather more likes and have lower lexical richness (cf. Table 4), which might actually indicate their attempt to mask illegitimate activities.

Shared Content. We then study the distributions of posts that are classified as "shared activity," i.e., originally made by another user, or articles, images, or videos linked from an external URL (e.g., a blog or YouTube). Figure 6 shows that baseline users generate more original posts, and share fewer posts or links, compared to like farm users.

Words per Post. Finally, Figure 7 plots the distributions of number of words that make up a text-based post, highlighting that posts of like farm users tend to have fewer words. Roughly half of the users in four of the like farms (AL-ALL, BL-USA, SF-ALL, and SF-USA) use 10 or less words in their posts, as opposed to 17 words by baseline users.

4.2 Lexical Analysis

Now we look at features that relate to the content of the timeline itself and consider a lexical analysis as a possible approach to differentiate regular timelines from malicious timelines involved into farm likes. While we only consider posts in English in our study, we note that similar lexical features could be extracted for different other languages (e.g. [32]. Interested readers might refer to [20] for more details).

Language. We start by analyzing the ratio of posts in English, i.e., for every timeline post, we filter out all non-English posts using an off-the-shelf language detection library.[3] For each user, we count the number of English-language posts and calculate its ratio with respect to the total number of posts. Figure 2 shows that the normal users and like farm users in USA (i.e., MS-USA, BL-USA, and AL-USA) mostly post in English, while users of worldwide campaigns (MS-ALL, BL-ALL, AL-ALL) have significantly fewer posts in English. For example, the median ratio of English posts for AL-ALL campaign is around 10% and that for SF-ALL around 15%.

Readability. We further analyze posts for grammatical and semantic correctness. We parse each post to extract the number of words, sentences, punctuation, non-letters (e.g., emoticons), and measure the lexical richness, as well as the Auto-

[3] https://python.org/pypi/langdetect

mated Readability Index (ARI) [21] and Flesch score [10]. Lexical richness, defined as the ratio of number of unique words to total number of words, reveals noticeable repetitions of distinct words, while the ARI [(4.71 × average word length) + (0.5 × average sentence length) - 21.43] gauges the comprehensibility of a text corpus. Table 4 shows a summary of the results. In comparison to like farm users, normal users post text with higher lexical richness (70% vs. 55%), ARI (20 vs. 15), and Flesch score (55 vs. 48), thus suggesting that normal users use a richer vocabulary and that their posts have higher readability.

4.3 Takeaway

Our analysis of user interaction with posts highlights several differences in both lexical and post interaction features of normal and like farm users. In the following section, we use these timelines features to detect like farm users using SVM.

5. DETECTING LIKE FARMS

Aiming to automatically distinguish like farm users from normal (i.e. baseline) users, we use a supervised two-class SVM classifier. From each baseline and like farm user's timeline, we extract four non-lexical interactive features and twelve lexical features, as explained in the previous section. Interactive features features are: average number of words per posts, average number of comments per posts, average likes per posts. The lexical features we use are: number of shared posts, number of characters, number of words, number of sentences, average word length, average sentence length, number of upper case letters, percentage of punctuations, percentage of numbers, percentage of non-letter characters, richness, ARI, and Flesch Score.

We form two classes by labeling like farm and baseline users' lexical features as positives and negatives, respectively. We use 80% and 20% of baseline and like farm users' lexical features from the labeled dataset to constitute the training and testing sets, respectively. We empirically find the appropriate values for γ, a parameter for *radial basis function kernel* [19], and υ, a parameter for SVMs by performing a greedy grid search on ranges $2^{-10} \leq \gamma \leq 2^{0}$ and $2^{-10} \leq \upsilon \leq 2^{0}$, respectively, on each training group.

Table 5 shows the effectiveness of our classifier with non-lexical Interactive features, i.e., users interactions with the posts. Note that, for each campaign, we train the classifier with the 80% non-lexical features from baseline and campaign training sets derived from the campaign users timelines. We use true positives, true negatives, and testing sets cardinalities to calculate precision, recall, and F1-measure. The poor classification performance for the stealthiest like farm (BL-USA) suggests that non-lexical features are not enough to accurately detect like farm users.

To improve the classification performance with the best possible timeline features, we evaluate the effectiveness of our classifier with lexical features. In the process, we filter out all users with no English-language posts (i.e, with R=0, see Figure 2). Likewise, we train the classifier with the 80% lexical features from baseline and like farm training sets. Table 6 shows the effectiveness of our classifier with only lexical features. We observe that our classifier achieves near perfect precision and recall for MS-USA, BL-USA, and AL-USA. Although its accuracy decreases by approximately 8% for SF-USA, in general the overall performance suggests that using lexical features is effective to automatically detect like farm users.

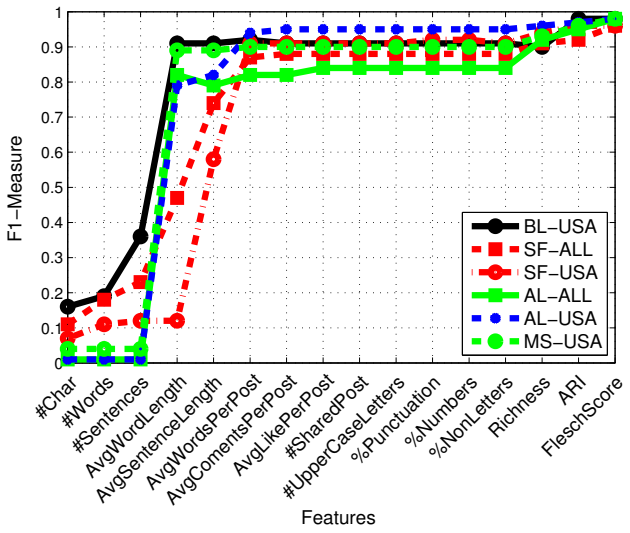


Figure 8: F1-measure for all lexical features measured for all campaigns. The X-axis shows the incremental inclusion of lexical features in both training and testing of SVM. Details of the classification performance for all lexical features are listed in Table 7.

Approximately 3% – 22% of like farm users do not have English language posts and are not considered in the analysis. To include them in the classification, we take all lexical and non-lexical features evaluate our classifier. We present the effectiveness of our classifier in Table 7. In this case, our classifier achieves high accuracy with F1-measure $\geq$ 96% for all like farms. Our results suggest the effectiveness of timeline features in detecting like farm users.

We further analyze the classification performance (in terms of F1-measure) to identify distinctive features among all lexical features. We incrementally add lexical and interaction-based features to train and test our classifier for all campaigns and report the performance in Figure 8. We observe that the average word length provides the most improvement in the F1-measure for all campaign except SF-USA. This suggests that, as compared to baseline users, like farm users use shorter words in their timeline posts. On the other hand, compared to baseline users, SF-USA users posts have shorter sentences.

Overall, our results demonstrate that it is possible to detect fake users from both sophisticated and naïve like farms once you incorporate additional profile information and in

Campaign	Average # Characters	Average # Words	Average # Sentences	Average Sentence Length	Average Word Length	Richness	ARI	Flesch Score
Baseline	4,477	780	67	6.9	17.6	**0.70**	**20.2**	55.1
AL-ALL	2,835	464	32	6.2	13.9	0.59	14.8	43.6
AL-USA	2,475	394	33	6.2	12.7	0.49	14.1	54.0
BL-USA	7,356	1,330	63	5.7	22.8	0.58	16.9	51.5
MS-USA	6,227	1,047	66	6.1	17.8	0.53	16.2	50.1
SF-ALL	1,438	227	19	6.3	11.7	0.58	14.1	45.2
SF-USA	1,637	259	22	6.3	12.0	0.55	14.4	45.6

Table 4: Lexical analysis of timeline posts of like farm users and normal users.

Campaign	Total Users	Training Set	Testing Set	TP	FP	TN	FN	Precision	Recall	Accuracy	F1-Measure
BL-USA	583	466	117	37	12	270	80	76%	32%	77%	45%
SF-ALL	870	696	174	139	9	273	35	94%	80%	90%	86%
SF-USA	653	522	131	110	5	277	21	96%	84%	94%	90%
AL-USA	827	662	164	113	4	278	51	97%	69%	88%	81%
AL-ALL	707	566	141	132	5	278	9	96%	94%	97%	95%
MS-USA	259	207	52	39	2	280	13	95%	75%	96%	84%

Table 5: Effectiveness of timeline based features (+SVM), interaction with timeline posts only, in detecting like farm users.

Campaign	Total Users	Training Set	Testing Set	TP	FP	TN	FN	Precision	Recall	Accuracy	F1-Measure
BL-USA	564	451	113	113	0	240	0	100%	100%	100%	100%
AL-ALL	675	540	135	129	2	238	6	98%	96%	98%	97%
AL-USA	570	456	114	113	0	240	1	100%	99%	99%	99%
SF-ALL	761	609	152	150	1	239	2	99%	99%	99%	99%
SF-USA	570	456	114	99	2	238	15	98%	87%	95%	92%
MS-USA	224	179	45	45	0	240	0	100%	100%	100%	100%

Table 6: Effectiveness of timeline based features (+SVM), lexical features only, in detecting like farm users.

Campaign	Total Users	Training Set	Testing Set	TP	FP	TN	FN	Precision	Recall	Accuracy	F1-Measure
BL-USA	583	466	117	113	1	281	4	99%	97%	99%	98%
SF-ALL	870	696	174	163	2	280	11	99%	94%	97%	96%
SF-USA	653	522	131	122	1	281	9	99%	93%	98%	96%
AL-USA	827	662	164	157	1	281	7	99%	96%	98%	97%
AL-ALL	707	566	141	137	1	281	4	99%	97%	99%	98%
MS-USA	259	207	52	50	0	282	2	100%	96%	99%	98%

Table 7: Effectiveness of timeline based features (+SVM), lexical and post interaction, in detecting like farm users.

particular timeline activities. Note that using a variety of feature sets, including lexical and post interaction features, makes it difficult for like farms to circumvent detection. For example, our results showed that like farm operators seem to rely on pre-defined lists of comments resulting in word repetition and lower lexical richness. It is challenging for like farm operators, who use automated scripts or cheap human labor in developing countries to orchestrate a large number of fake accounts, to match the diversity and richness of real users' timeline posts.

6. RELATED WORK

Prior work has focused on the analysis and the detection of fake accounts in online social networks [2, 4, 31, 11, 30]. In this paper, we specifically focus on detecting fake accounts that are employed by like farms to boost the number of Facebook page likes. In the following, we provide a summary of related research and discuss how our methods and findings differ from related work.

In [25, 16], researchers employed a honeypot-based approach to harvest accounts used in spamming and build a classifier. Specifically, they created passive honeypot accounts on MySpace and Twitter, and monitored friend/follow requests and messages to identify spamming accounts. In contrast, our work is based on Facebook honeypot pages that actively engaged like farms to provide (paid) page likes. Also, [25] and [16] did not leverage timeline-based features, which we use in our work for classification.

[29] studied human involvement in Weibo's reputation manipulation services. They showed that simple evasion attacks (e.g., workers modifying their behavior) as well as poisoning attacks (e.g., administrators tampering with the training set) can severely affect the effectiveness of machine learn-

ing algorithms to detect malicious crowd-sourcing workers. Partially informed by their work, we do not only cluster like activity performed by users but also use timeline information to detect like farm accounts.

Also, other studies have analyzed services that sell *Twitter followers* [24], traffic fake and compromised Twitter accounts [27], and *crowdturfing* in Online Social Networks [23]. Specific to Facebook fraud is the work by [1], who introduced CopyCatch, a technique currently deployed by Facebook to detect fraudulent accounts. Authors showed that CopyCatch can detect fake likes by identifying groups of connected users liking a set of pages within a short time frame. SynchroTrap [5] extended CopyCatch by clustering accounts that perform similar, possibly malicious, synchronized actions. Like CopyCatch, SynchroTrap is currently deployed, allowing Facebook engineers to tune parameters such as time-window and similarity thresholds in order to improve detection accuracy. However, results from [8], who also used honeypot-based measurements of like farms, highlighted the presence of stealthier like farms, exhibiting behavior that may be challenging to effectively detect with tools like CopyCatch and SynchroTrap. In fact, while some farms seem to be operated by bots (producing large bursts of likes and having limited numbers of friends), not really trying to hide the nature of their operations, others actually aim to mimic regular users' behavior to evade detection. In this paper, we take a significant step further and empirically demonstrate that it is indeed the case. Our evaluation of graph co-clustering techniques shows that stealthy like farms successfully evade detection by avoiding lockstep behavior and liking sets of seemingly random pages. We also use the lexical and post interaction features of user timelines in order to build a classifier that allows us to detect like farm users with high accuracy.

Other methods have also been used in the past to detect fake and compromised accounts, such as using unsupervised anomaly detection techniques to distinguish malicious behavior from normal to detect compromised, fake, and colluding accounts [28], as well as detecting lockstep behavior using temporal features [13, 14]. More recently, [26] investigated the use of mappings of accounts to IP addresses to detect malicious accounts. Our work complements these approaches.

7. CONCLUDING REMARKS

Detecting fraudulent accounts on social networks and forums is crucial to maintain confidence among legitimate users and investors. This paper focused on the detection of users from like farms on Facebook, i.e., paid services artificially boosting the number of likes on a given Facebook page. We crawled and gathered both liking pattern and timeline activities from like farms accounts as well as a baseline of normal users. Based on these liking patterns, we evaluated the effectiveness of existing graph based fraud detection algorithms, such as CopyCatch [1] and SynchroTrap [5], in detecting like farm accounts. Our results showed that sophisticated like farms can successfully evade the existing algorithms. Thus, we unveiled, via a clustering analysis, a potential counter-attack to these state-of-the-art detection tools as accounts from same like farms like very popular pages or relatively niche pages.

Aiming to address this problem, we set to incorporate additional profile information from accounts' timelines, and investigated whether this helps to improve detection of like farms' activities. First, we used a latent semantic analysis approach, and a used machine learning classifier to distinguish normal user from like farm users, albeit, with relatively poor performance (which can be partly explained by the short nature of the posts). Then, we analyzed available timeline information, identifying textual and activity (interaction) features that enable an accurate distinction between like farm and normal users. In particular, we found that posts made by like farm accounts have 43% fewer words, a more limited vocabulary, and lower readability than normal users' posts. Moreover, like farm posts were highly targeted to some specific topics, generated significantly more comments and likes, and a large fraction of their posts consists of non original and often redundant "shared activity" (i.e., repeatedly sharing posts made by other users, articles, videos, and external URLs). Finally, by leveraging these lexical and interaction features, we experimented with machine learning algorithms to detect like farms accounts with high accuracy.

8. REFERENCES

[1] A. Beutel, W. Xu, V. Guruswami, C. Palow, and C. Faloutsos. CopyCatch: Stopping Group Attacks by Spotting Lockstep Behavior in Social Networks. In *WWW*, 2013.

[2] Y. Boshmaf, D. Logothetis, G. Siganos, J. R. Leria, J. Lorenzo, M. Ripeanu, and K. Beznosov. Integro: Leveraging Victim Prediction for Robust Fake Account Detection in OSNs. In *NDSS*, 2015.

[3] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML/PKDD LML Workshop*, 2013.

[4] Q. Cao, M. Sirivianos, X. Yang, and T. Pregueiro. Aiding the Detection of Fake Accounts in Large Scale Social Online Services. In *NSDI*, 2012.

[5] Q. Cao, X. Yang, J. Yu, and C. Palow. Uncovering Large Groups of Active Malicious Accounts in Online Social Networks. In *CCS*, 2014.

[6] B. Carter. *The Like Economy: How Businesses Make Money with Facebook*. QUE Publishing, 2013.

[7] T. Chen, M. A. Kaafar, A. Friedman, and R. Borelli. Is More Always Merrier? A Deep Dive Into Online Social Footprints. In *WOSN*, 2012.

[8] E. De Cristofaro, A. Friedman, G. Jourjon, M. A. Kaafar, and M. Z. Shafiq. Paying for likes? understanding facebook like fraud using honeypots. In *ACM IMC*, 2014.

[9] Facebook, Inc. Form 10-K. http://investor.fb.com/secfiling.cfm?filingID=1326801-15-6, January 2015.

[10] R. Flesch. A new readability yardstick. *Journal of Applied Psychology*, 32(3):221–233, 1948.

[11] H. Gao, J. Hu, C. Wilson, Z. Li, Y. Chen, and B. Y. Zhao. Detecting and Characterizing Social Spam Campaigns. In *IMC*, 2010.

[12] A. Hogenboom, F. Frasincar, F. de Jong, and U. Kaymak. Using Rhetorical Structure in Sentiment Analysis. *Communications of the ACM*, 58(7):69–77, June 2015.

[13] M. Jiang, P. Cui, A. Beutel, C. Faloutsos, and S. Yang. CatchSync: Catching Synchronized Behavior in Large Directed Graphs. In *KDD*, 2014.
[14] M. Jiang, P. Cui, A. Beutel, C. Faloutsos, and S. Yang. Inferring Strange Behavior from Connectivity Pattern in Social Networks. In *PAKDD*, 2014.
[15] Y. Kluger, R. Basri, J. T. Chang, and M. Gerstein. Spectral Biclustering of Microarray Data: Co-Clustering Genes and Conditions. *Genome Research*, 13:703–716, 2003.
[16] K. Lee, J. Caverlee, and S. Webb. Uncovering Social Spammers: Social Honeypots + Machine Learning. In *SIGIR*, 2010.
[17] K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf. An introduction to kernel-based learning algorithms. volume 12, pages 181–201, 2001.
[18] G. Salton and M. J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA, 1986.
[19] B. Schölkopf, J. C. Platt, J. C. Shawe-Taylor, A. J. Smola, and R. C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7):1443–1471, July 2001.
[20] M. Silberztein. The lexical analysis of natural languages. *Finite-state language processing*, pages 176–205, 1997.
[21] E. A. Smith and R. J. Senter. Automated Readability Index. AMRL-TR-66-220, http://www.dtic.mil/dtic/tr/fulltext/u2/667273.pdf, 1967.
[22] B. Snyder. Facebook added 10 million small business pages in a year. http://fortune.com/2015/04/30/facebook-small-business, April 2015.
[23] J. Song, S. Lee, and J. Kim. Crowdtarget: Target-based detection of crowdturfing in online social networks. In *CCS*, 2015.
[24] G. Stringhini, M. Egele, C. Kruegel, and G. Vigna. Poultry Markets: On the Underground Economy of Twitter Followers. In *WOSN*, 2012.
[25] G. Stringhini, C. Kruegel, and G. Vigna. Detecting Spammers on Social Networks. In *ACSAC*, 2010.
[26] G. Stringhini, P. Mourlanne, G. Jacob, M. Egele, C. Kruegel, and G. Vigna. Evilcohort: detecting communities of malicious accounts on online services. In *Usenix Security Symposium*, 2015.
[27] K. Thomas, D. McCoy, C. Grier, A. Kolcz, and V. Paxson. Trafficking Fraudulent Accounts: The Role of the Underground Market in Twitter Spam and Abuse. In *Usenix Security Symposium*, 2013.
[28] B. Viswanath, M. A. Bashir, M. Crovella, S. Guha, K. P. Gummadi, B. Krishnamurthy, and A. Mislove. Towards Detecting Anomalous User Behavior in Online Social Networks. In *Usenix Security Symposium*, 2014.
[29] G. Wang, T. Wang, H. Zheng, and B. Y. Zhao. Man vs. Machine: Practical Adversarial Detection of Malicious Crowdsourcing Workers. In *Usenix Security Symposium*, 2014.
[30] C. Yang, R. Harkreader, J. Zhang, S. Shin, and G. Gu. Analyzing Spammers Social Networks for Fun and Profit. In *WWW*, 2012.
[31] Z. Yang, C. Wilson, X. Wang, T. Gao, B. Y. Zhao, and Y. Dai. Uncovering Social Network Sybils in the Wild. In *ACM IMC*, 2011.
[32] H.-P. Zhang, H.-K. Yu, D.-Y. Xiong, and Q. Liu. HHMM-based Chinese Lexical Analyzer ICTCLAS. In *Workshop on Chinese Language Processing*, 2003.