Uncovering Fake Likers in Online Social Networks

Prudhvi Ratna Badri Satya[†] Kyumin Lee[†] Dongwon Lee[§]
Thanh Tran[†] Jason (Jiasheng) Zhang[§]

[†] Department of Coputer Science, Utah State University, UT, USA
[§] College of Information Sciences and Technology, The Pennsylvania State University, PA, USA

{prudhvibadri|thanh.tran}@aggiemail.usu.edu kyumin.lee@usu.edu {dongwon|jpz5181}@psu.edu

ABSTRACT

As the commercial implications of *Likes* in online social networks multiply, the number of *fake Likes* also increase rapidly. To maintain a healthy ecosystem, however, it is critically important to prevent and detect such *fake Likes*. Toward this goal, in this paper, we investigate the problem of detecting the so-called "*fake likers*" who frequently make *fake Likes* for illegitimate reasons. To uncover *fake Likes* in online social networks, we: (1) first collect a substantial number of profiles of both fake and legitimate *Likers* using linkage and honeypot approaches, (2) analyze the characteristics of both types of *Likers*, (3) identify effective features exploiting the learned characteristics and apply them in supervised learning models, and (4) thoroughly evaluate their performances against three baseline methods and under two attack models. Our experimental results show that our proposed methods with effective features significantly outperformed baseline methods, with accuracy = 0.871, false positive rate = 0.1, and false negative rate = 0.14.

Keywords: Fake Likers; Fiverr; Microworkers; Online Social Networks; Facebook

1. INTRODUCTION

Everyday billions of people actively use online social networks (OSNs) such as Facebook, Twitter, and Instagram. They share information regarding their daily lives, engage with friends and followers, and express their opinions through reviews, votes, and *Likes* in OSNs. Similarly, businesses promote and attempt to increase positive images of their services and products in OSNs. In such OSNs, in recent years, the commercial implications of *Likes* have multiplied. For instance, the *Like* function can be a good marketing tool since businesses can recognize interested users and the accumulated *Likes* help promote businesses in various searches. Similarly, *Likes* help understand users better in OSNs [8].

To benefit from the increased importance of *Likes*, however, a set of new markets to buy and sell illegitimate *Likes* also have emerged in OSNs. In such markets, for instance, by paying $5, it is fairly easy to buy thousands of *Likes*. Note that such paid liking activities are strictly prohibited in OSNs (e.g., on Facebook[1]). In this paper, we refer to such illegitimate *Likes* as **fake Likes** . As such artificially manipulated fake *Likes* hurt the healthy ecosystem in OSNs, it is important to actively prevent, deter, detect, and respond to them. In particular, in this paper, we focus on the problem of "detecting fake *Likes*". Similar problems including detecting fake reviews, spammers/sybils, fake followers, or manipulated keyword searches have been reported in literature (e.g., [12, 13, 14, 16, 18]).

While related, however, we believe that the problem of detecting *fake Likes* is more challenging because: (1) information available in a *Like* is much more limited than fake reviews or sybils, and (2) the difference between fake and legitimate *Likes* is often unclear for 3rd party to judge. Therefore, as a way to detect *fake Likes*, we propose to first detect "fake likers" who have performed at least k *fake Likes* in OSNs. As k increases, the likelihood of one being a *fake Liker* increases but the number of *fake Likers* in a dataset decreases. As such, to maintain a proper data size for our experiments, k is set to 2. Formally, we solve the following problem:

Problem 1 (Fake Liker Detection Problem) Consider a user class with binary types C={`fake-liker`, `legit-liker`}. Then, for a user u, given a set of u's features $f \in F$, and a training set S of users labeled with C, learn a classifier γ such that: $\gamma : u \rightarrow C$. ∎

Recently, researchers (e.g., [1, 3, 17]) have studied the related problem. Despite promising results, however, their approaches are limited to some extent as they require: (1) expensive features such as temporal data of *Likes*, and (2) an array of input parameters with tuning. In complementing these prior works and improving shortfalls therein, in this paper, we make the following contributions: (1) Collecting over 13,000 fake and legitimate *Likers* in Facebook from two types of sources–Fiverr and Microworkers–using both linkage and honeypot methods, we present comprehensive analysis to understand both types of *Likers* better; (2) Based on the learned characteristics of both types of *Likers*, then, we identify five types of effective and cost-effective features and apply them to build supervised learning models; and (3) Against three baseline methods under individual and coordinated attack models, finally, we empirically validate the

[1] https://www.facebook.com/10152309368645766/

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CIKM'16 , October 24-28, 2016, Indianapolis, IN, USA

© 2016 ACM. ISBN 978-1-4503-4073-1/16/10. . . $15.00

DOI: http://dx.doi.org/10.1145/2983323.2983695

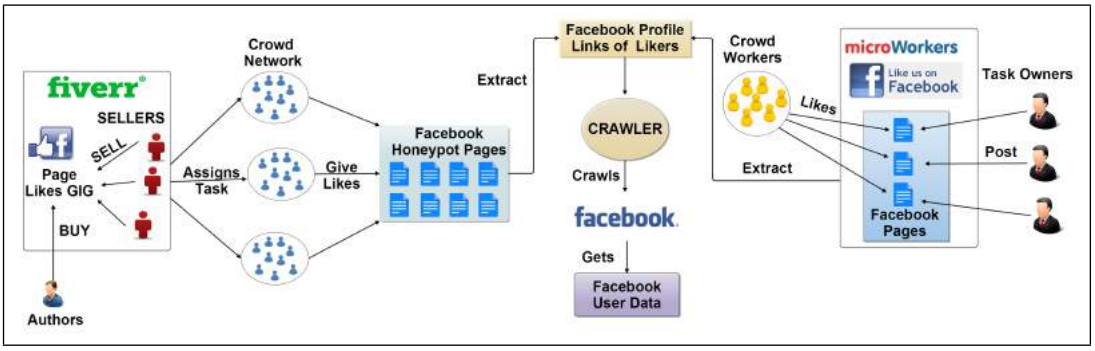


Figure 1: Collecting *fake Likers* from Fiverr and Microworkers.

effectiveness of our proposed solutions with high accuracies and low false positive rates.

2. RELATED WORK

Researchers [9, 15] deployed honeypots to uncover spammers on Twitter and Facebook, and developed machine learning based classification models to detect spammers. Lee et al. [11] performed comprehensive analysis of Fiverr to reveal the existence of crowdturfing tasks therein, and developed a crowdturfing task detection method to remove such tasks.

De Cristofaro et al. [5] presented a comparative study of Facebook ads and *Likes* farms by analyzing demographic, temporal, and social characteristics of likers but did not investigate the detection of *fake Likers* accounts. Viswanath et al. [17] used the principal component analysis technique to detect anomalous accounts on Facebook. While showing good performance, this work lacks comprehensive analysis on the differences between the users of various *Like* platforms, and temporal features that they used are expensive to obtain. In our work, however, we aim to achieve the accurate detection without expensive temporal features.

CopyCatch [1] detected undesirable accounts making *fake Likes* by extracting near bipartite cores from users and Facebook pages based on *Likes* and liked time. SynchroTrap [3] detected groups of malicious accounts by running hierarchical clustering based on their liking similarity during the same time interval. However, in our study, we found that Facebook security system that reportedly uses both methods did not detect *fake Likes* (that we detected), and did not remove their accounts even two months later.

3. DATA COLLECTION

To build an accurate classification model, the first challenge is to obtain a good size of labeled training data. As it is virtually impossible for 3rd party to tell if a *Like* is "fake" or not, instead, we actively solicit *fake Likes* in two crowdsourcing platforms (e.g., Fiverr and Microworkers), and collect the Facebook users who gave at least k *fake Likes*–thus by definition *fake Likers*. Recent research [10, 11] reported that some users in Fiverr and Microworkers have performed malicious tasks (e.g., fake liking or following). Figure 1 illustrates our data collection process. Next, as positive samples, we also collected two legitimate user sets – conference groups and a random pool. In total, we constructed two fake and two legitimate users sets with 6,895 faker liker profiles and 6,253 legitimate liker profiles, as summarized in Table 1. We describe further details in data collection below.

Figure 2: A Facebook *Like* ad by a seller in Fiverr.

Types	|Likers|	|pages|	|posts|	|friends|
Fake (Fiverr)	3,207	1.86M	0.21M	0.82M
Fake (Microworkers)	3,688	1.63M	0.33M	1.54M
Legit (Conference)	2,552	0.57M	0.17M	0.87M
Legit (Random)	3,701	0.60M	0.28M	2.24M
Total	13,148	4.66M	0.99M	5.47M

Table 1: Sizes of labeled sets.

Fake Likers in Fiverr. Fiverr is a seller-driven marketplace. As shown in Figure 2, for instance, a seller advertises to sell hundreds of *fake Likes* for a fee. Such a seller often acts as a broker, who in turn hires hundreds of other *fake Likers* to actually make *fake Likes*. We contacted ten sellers in Fiverr to purchase *fake Likes* to our honeypot Facebook pages. To avoid receiving legitimate *Likes* from other Facebook users, our honeypot pages clearly displayed a message "This is a fake page. Please do not like this" and had no other contents. Then, at each hour for next 10 days, a crawler attempted to collect all user related information who "liked" the honeypot pages. At the end, out of 3,916 *Likes* made to the honeypot pages, 3,207 users have made at least k *Likes* (e.g., $k = 2$) and had no privacy setting. Then, we collected 3,207 profiles of those *fake Likers* including a total of 1.86 million pages that they liked, 0.21 million posts, and 0.82 million friends information in Facebook.

Fake Likers in Microworkers. In contrast to Fiverr, Microworkers is a buyer-driven platform where a buyer creates an ad (e.g., "I'll pay $0.01 if you *Like* a Facebook page XYZ"). First, we selected 353 tasks in Microworkers, whose titles contained "Facebook Like" and extracted targeted Facebook page URLs therein. Then, we extracted a list of users who liked the targeted pages. These are potential *fake Lik-*

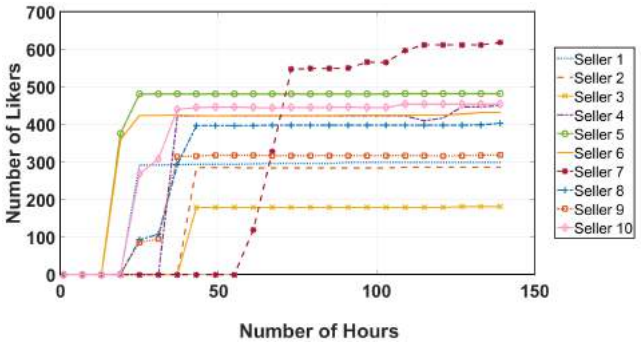


Figure 3: Temporal change of # of *fake Likes*.

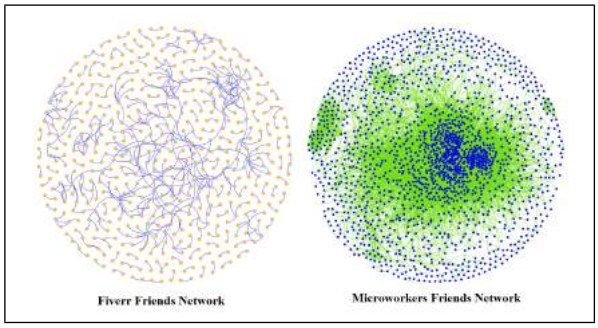


Figure 4: The 2-hop (mutual) friendship relations in Fiverr and Microworkers communities.

ers. Then, we selected 3,688 users who liked at least k pages among 353 target pages (e.g., $k = 2$) and did not have any privacy setting, and collected their profiles including a total of 1.63 million pages that they liked, 0.33 million posts, and 1.54 million friends information in Facebook.

Legitimate Likers from Conferences. As the first heuristic to collect legitimate *Likers*, we borrowed the idea from [17], and collected 2,552 user profiles from 13 CS conference groups on Facebook. As in [17], we also assumed that these technically savvy users were less likely to be infected by malware or other attacks, and unlike to perform fake liking activity. In addition, we sampled 1,000 out of the 2,552 user profiles, checked whether they were suspicious accounts or not, and found none. These users liked 0.57 million pages, posted 0.17 million posts, and had 0.87 million friends.

Legitimate Likers from a Random Pool. Finally, from 20 random seed users who live in different countries, we crawled their 2-hop friends network using the breadth first search (BFS) technique. From the 28,200 users in the network, to further randomize, we randomly selected 5,700 users. Out of 5,700 random users, 3,779 users did not have any privacy setting. Then, two labelers conducted manual labeling process for the 3,779 user profiles, investigating each user's posts and timeline information. They achieved 99.7% agreement with Kappa coefficient of 0.78. Selecting only users agreed by both labelers, at the end, we kept 3,701 legitimate users who liked 0.6 million pages, posted 0.28 million posts, and had 2.24 million friends. Note that almost 2% accounts on Facebook are undesirable accounts according to Facebook SEC filing report [6].

Download. Our dataset is publicly available at:

http://digital.cs.usu.edu/~kyumin/data.html

4. UNDERSTANDING LIKERS

In this section, we analyze the characteristics of both fake and legitimate *Likers*.

Who are Fake Likers? First, we analyzed how quickly each Fiverr seller delivered *fake Likes*. Figure 3 illustrates the temporal change of the number of *Likes* by ten sellers. Note that most of *fake Likes* were delivered within three days in a bursty fashion, that is consistent with the finding in [5].

Next, we analyzed the demographics of *fake Likers* associated with Fiverr sellers. Since we created honeypot pages on Facebook, we were able to access the demographic information of the *fake Likers*. 73% of all *fake Likers* were from 10 countries–Egypt, Iraq, Tunisia, Algeria, Morocco, Philippines, Brazil, Jordan, Saudi Arabia, UK–which are all developing countries except UK. We also observed most *fake Likers* were in the range of 18-34 years old, regardless of sellers whom they were associated with. Surprisingly, some *fake Likers* were teenagers, and overall, there were more male than female *fake Likers*.

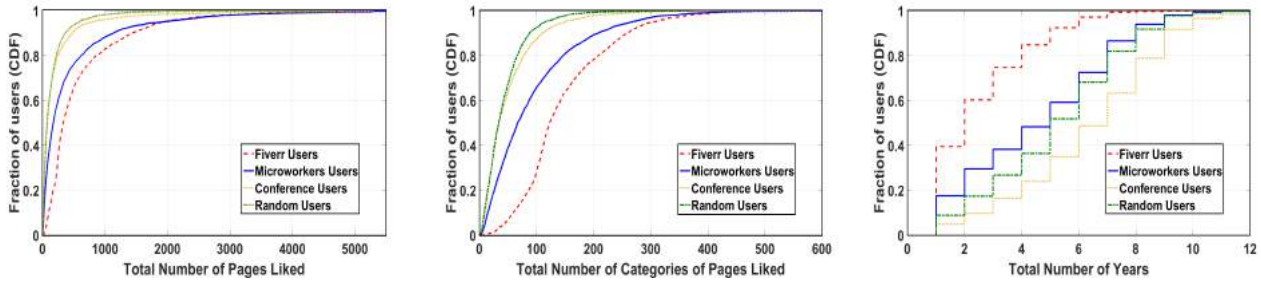
Another interesting question is if *fake Likers* unliked our honeypot pages a certain period later (e.g., a week or a month)? Researchers observed such behavior on Twitter that spammers often followed and unfollowed users [9]. In our study, however, only 10% of *fake Likers* unliked our pages when we checked the status after 3 months.

Fiverr vs. Microworkers Fake Likers. First, we analyzed how *fake Likers* from Fiverr and Microworkers were connected via Facebook friendship network and observed no direct friendship connections among *fake Likers* in Fiverr. However, *fake Likers* in Microworkers were densely connected. For instance, on average, 1,099 out of 3,688 *fake Likers* (i.e., nodes) in Microworkers were connected via 5,239 friendship connections (i.e., edges) with 9.5 degree and 0.446 clustering coefficient on average.

In Figure 4, a node indicates a *fake Liker* and if a pair of *fake Likers* within a community has at least one common friend (i.e., 2-hop friend), we added an edge between them. *fake Likers* in Fiverr were sparsely connected whereas *fake Likers* in Microworkers were densely connected. In particular, 570 *fake Likers* in Fiverr were connected via 578 friendship connections with 2.0 degree and 0.233 clustering coefficient on average while 1,802 *fake Likers* in Microworkers were connected via 48,173 friendship connections with 53.4 degree and 0.612 clustering coefficient on average. In addition, we found no direct friendship relations between *fake Likers* of Fiverr and Microworkers. However, there were 672 mutual friends between *fake Likers* from Fiverr and Microworkers.

Next, we computed pair-wise similarities of pages liked by *fake Likers* to find out any cross-workers between Fiverr and Microworkers who performed tasks in both sites. We observed only a negligible similarity due to verified pages a.k.a authentic pages (e.g., Amazon, Disney). This implies that there were few cross-workers. However, a pairwise similarity of pages liked by *fake Likers* from each site/community was much larger.

Next, we analyzed what URLs *fake Likers* frequently shared or posted on Facebook. Table 2 presents top 5 URLs in each platform. For instance, *fake Likers* in Fiverr posted URLs re-



(a) Number of liked pages. (b) Number of liked page categories. (c) Longevity (years).

Figure 5: The CDFs of fake and legitimate *Likers*.

Platform	Top 5 URLs	# of Likers
Fiverr	https://www.facebook.com/facebook/	215
	http://apps.facebook.com/monsterlegends	89
	http://apps.facebook.com/dragoncity	38
	http://apps.facebook.com/stick_run	37
	http://apps.facebook.com/topeleven	31
Microworkers	https://www.krowdster.co	537
	http://www.indiegogo.com/projects/1240265	412
	https://www.indiegogo.com/projects/1528609	312
	http://www.fitneszszakuzlet.hu	276
	https://www.mykomms.com/	197

Table 2: Top 5 URLs shared or posted by *fake Likers*, and corresponding # of *Likers*.

lated to Facebook apps and games while those in Microworkers posted URLs related to advertisements of crowdfunding projects and other websites. This may indicate that Microworkers workers used their Facebook accounts for not only fake liking activities, but also other crowdturfing tasks.

Based on the analysis, we conclude that *fake Likers* in Fiverr are distinctively different from those in Microworkers in terms of direct relationship, mutual friends, job tasks that they performed, and URL sharing patterns.

Fake vs. Legitimate Likers. Next, we contrasted fake and legitimate *Likers*. Figure 5(a) clearly shows that *fake Likers* from Fiverr and Microworkers performed more page liking activities than legitimate likers. In particular, 90% of random and conference legitimate users liked at most 369 and 481 pages, respectively, whereas 90% of *fake Likers* from Fiverr and Microworkers liked at most 1,481 and 1,137 pages, respectively. In addition, Figure 5(b) presents the number of categories associated with Facebook pages liked by fake and legitimate *Likers*. Note that *fake Likers* liked pages with more diverse categories than legitimate *Likers*. In particular, Fiverr's *fake Likers* liked pages with the most diverse categories. This may reflect the lack of personal interests of *fake Likers* in liking pages as they would like any pages for a fee. Finally, Figure 5(c) shows that the accounts of legitimate *Likers* were created earlier than those of *fake Likers*. In particular, we note that conference users created their accounts earlier than other users, and Fiverr's accounts were created recently (i.e., 80% of them were created within four years).

5. FEATURE ENGINEERING

Based on our findings from Section 4, now, we present four sets of features effective to detect *fake Likers* as follows:

1. **Profile Features**: # of lines in *About* section; longevity of an account; # of friends.

2. **Posting Activity Features**: These include two types of posts by a user u–u's own posts (i.e., posts created by u) and shared posts (i.e., posts created by other user but shared by u). Posts include photos, pages, videos, text messages, etc. We extracted following features: average # of posts per day; total # of posts created by u; proportion of shared photos, posts or pages out of total # of posts; maximum # of posts in a day; average # of URLs per post; and skewness of daily posted posts.

3. **Page Liking Features**: We extracted two features: (1) category entropy: A larger category entropy indicates that a user randomly liked pages under various categories. Given a list of Facebook categories $C = c_1, c_2, c_3, \ldots, c_k$ and corresponding number of pages in each category liked by a user u, the user's category entropy is calculated as follows: $CatEntropy(u) = -\sum_{i=1}^{k} \frac{p_i}{N} \log \frac{p_i}{N}$, where N is the total # of pages liked by u, and p_i is # of liked pages under a category i; and (2) proportion of verified pages out of total # of pages liked by a user.

4. **Social Attention Features**: average # of *Likes* (selected by other users) per post; average # of comments per post; and average # of shared per post.

In addition, for us to implement three state-of-the-art approaches (i.e., PCA [17], SynchroTrap [3] and CopyCatch [1]) for comparison, we also extracted temporal features. We collected temporal snapshot data of 1,400 *Likers* (i.e., 700 fake and 700 legitimate *Likers*) and extracted following two features: (1) A change rate of # of liked pages during 30 days: We extracted 30 values, each of which was # of liked pages by a user of the day. Then, we measured a standard deviation of those 30 values; and (2) A change rate of category entropies during 30 days: We measured a category entropy per day and computed a standard deviation of those 30 category entropies.

To avoid the features that are too similar or correlated, we measured Pearson correlation and computed Chi-square values for the four feature sets. Note that we did the same process for all feature sets including temporal features in a small dataset containing temporal data, but showed Pearson correlation results for the four feature sets in the entire dataset in this paper. The largest correlation score was less than 0.5, so we kept all the features.

Features	F. Likers	L. Likers
Category entropy	6.35	4.47
Longevity	3.62	5.53
Average # of posts per day	0.31	0.12
# of lines in *About* section	4.02	4.08
Proportion of verified pages	0.23	0.26

Table 3: Top 5 features and average feature values.

Finally, we computed the Chi-square values [20] to rank features in the order of the largest distinguishing power. The larger Chi-square value a feature has, the larger distinguishing power it has. Table 3 shows top-5 results, and average feature values of fake and legitimate *Likers*.

6. EMPIRICAL VALIDATION

In this section, we conduct three experiments: (1) we build *fake Liker* classifiers in a small dataset with temporal data to evaluate against three baseline methods (i.e., PCA [17], SynchroTrap [3] and CopyCatch [1]); (2) we build *fake Liker* classifiers in the entire dataset without temporal data, and compare the classifiers with the three baseline methods; and (3) we measure how robust our classification models are under two attack scenarios–individual attack model and coordinated attack model.

For evaluation, we measured accuracy, false positive rate (FPR), and false negative rate (FNR). In our context, a false positive indicates a legitimate user misclassified as *fake Liker*, while a false negative is a *fake Liker* misclassified as a legitimate user. Note that it is critical for our solutions to yield few false positives even if they yield some false negatives.

Detecting Fake Likers in a Small Dataset. First, out of 13,148 Facebook *Likers* in our dataset in Table 1, we further collected temporal data of 1,400 (i.e., 700 fake and 700 legitimate *Likers*) for 30 days between Dec 1 and Dec 30, 2015 because the PCA based approach in [17] required temporal features. Then, we split this small dataset to training and test sets. The training set consisted of 1,000 *Likers* (i.e., 500 *fake Likers* and 500 legitimate *Likers*), and the test set consisted of 400 *Likers* (i.e., 200 *fake Likers* and 200 legitimate *Likers*).

In our classification approach, we built classifiers based on 30+ machine learning algorithms (e.g., LogitBoost [7], Random Forest [2], XGBoost [4], SVM) using our proposed features to check which classifier produced the best result.

For PCA based approach [17], we extracted 30 temporal, 1,079 spatial/categorical and 30 spatio-temporal/category-temporal feature values from each *Liker*'s profile in both training and test sets as the authors did. We determined principal components and observed 95% variance in the top 400 principal components. We computed the L2 norm [19] and set the squared prediction error (SPE) as a threshold to find *fake Likers*. If a user exceeds the SPE value, then the user is likely to be a *fake Liker*. For determining the SPE, we changed a threshold value from 1% to 99% by increasing 1% each time in the training set. We found the optimal threshold, and applied the threshold value to the test set.

SynchroTrap [3] measures page liking similarity of each pair of users, and runs single-linkage hierarchical clustering. The output of the algorithm is a dendrogram structure, and the algorithm requires a cutoff threshold (i.e., similarity threshold) and a minimum size of a cluster to determine clusters of malicious accounts. Again, we found the best cutoff and size values from the training set and applied to the test set.

Approach	Accuracy	FPR	FNR
PCA	0.690	0.30	0.32
SynchroTrap	0.505	0	0.99
CopyCatch	0.565	0.85	0.02
PCA - Test	0.690	0.30	0.32
SynchroTrap - Test	0.635	0	0.73
CopyCatch - Test	0.655	0.46	0.23
our LogitBoost	0.875	0.11	0.13
our Random Forest	0.885	0.09	0.13
our XGBoost	**0.897**	0.08	0.11

Table 4: Experimental results in a small dataset containing temporal data.

Approach	Accuracy	FPR	FNR
PCA	0.563	0.49	0.38
SynchroTrap	0.521	0	0.91
CopyCatch	0.601	0.80	0.04
PCA - Test	0.576	0.28	0.54
SynchroTrap - Test	0.620	0.01	0.71
CopyCatch - Test	0.669	0.09	0.55
our LogitBoost	0.856	0.16	0.12
our Random Forest	0.865	0.15	0.11
our XGBoost	**0.871**	0.10	0.14

Table 5: Experiment results in the entire dataset.

CopyCatch [1] discovers groups of *fake Likers* by measuring page liking similarity of the groups in a specific time range, and outputs near bipartite cores/graphs which can be considered as malicious accounts. This method needs three input parameters: (1) minimum number of users in a near bipartite core; (2) minimum number of pages in a near bipartite core; and (3) how densely users are connected to pages (e.g., each user in a near bipartite core should like 90% pages). We varied these parameter values and found the best values from the training set and applied them to the test set.

Table 4 presents experimental results of the three baseline methods and our three most effective classifiers. XGBoost classifier achieved 0.897 accuracy, 0.08 FPR and 0.11 FNR, improving up to 0.392 (= 0.897 - 0.505) accuracy compared with the baseline methods. PCA, SynchroTrap and CopyCatch achieved 0.690, 0.505 and 0.565 accuracy, respectively. Their upper bound results (i.e., PCA - Test, SynchroTrap - Test and CopyCatch - Test which found optimal threshold/parameter values within the test set) achieved 0.690, 0.635 and 0.655 accuracy, respectively. These two different results show that a weakness of the baseline methods is hard to find optimal threshold or input parameter values. Second, even though we found the upper bound, they were still less effective than our classification models. Interestingly, SynchroTrap achieved 0 FPR but only identified 1% of *fake Likers* (i.e., 0.01 recall).

Detecting Fake Likers in the Wild. Now we turn to detect *fake Likers* in the entire dataset containing the profiles of 13,148 users without using expensive temporal data/features. In this experiment, we conducted 10-fold cross-validation, creating 10 pairs of training and test sets. All three baselines were applied to proposed 16 features excluding temporal features. Similar to what we did in the previous experiment, PCA, CopyCatch and SynchroTrap found an optimal threshold in each training set and applied it to each test

Features	Accuracy	FPR	FNR
Social attention features (3 feat.)	0.861	0.12	0.15
Profile features (3 feat.)	0.842	0.13	0.17
Posting activity features (8 feat.)	0.841	0.13	0.17
Page liking features (2 feat.)	0.824	0.17	0.17

Table 6: Our classification results under the coordinated attack model.

set. In addition, we measured the upper bound of the three baseline methods.

Table 5 show experimental results in the entire dataset. Again, XGBoost based classifier outperformed PCA, SynchroTrap and CopyCatch, achieving 0.871 accuracy, 0.1 FPR and 0.14 FNR. We further focused on misclassification cases of XGBoost. Out of 1,013 false negatives, 56 *fake Likers* were from Fiverr and the remaining 957 were from Microworkers. This shows Microworkers workers used their own personal facebook accounts for fake-liking activity unlike Fiverr. In false positive cases, some legitimate *Likers* had similar behaviors with *fake Likers* by posting many posts in a single day and having higher proportion of shared pages out of the total number of posts.

We also evaluated the performance of our approach in an unbalanced dataset consisting of 2% *fake Likers* [6] and 98% legitimate *Likers*. Our approach consistently outperformed the three baseline methods.

Robustness of Our Approach. To measure the robustness of our approaches, next, we simulated two attack models: (1) individual attack model; and (2) coordinated (group) attack model.

In the individual attack model, we assumed that each *fake Liker* *independently* selects one of our features, and then change its value to a legitimate *Liker*'s feature value. Specifically, given a range of feature values of legitimate *Likers*, the simulator randomly choose a value which is used for a *fake Liker*. We conducted 10-fold cross-validation and ran the simulation 10 times. Our XGBoost based classifier achieved 0.855 accuracy, 0.157 FPR and 0.133 FNR, decreasing 0.016 accuracy compared with our approach (0.871 accuracy).

In the coordinated attack model, we assumed that all *fake Likers* choose the *same* feature or features, and change its value or their values to a legitimate *Liker*'s feature value or values. Compared with the individual attack model, these *fake Likers* already know which feature/features they are going to manipulate. We tested changing one feature to multiple features. Again, we conducted 10-fold cross-validation and ran 10 times.

Table 6 show experimental results under the coordinate attack model. The coordinated attack for single feature decreased accuracy to between 0.834 and 0.870. When the fake likers targeted all features in one of four feature sets, accuracy decreased to between 0.824 and 0.861. The coordinated attack model slightly affected the performance of our approach (0.871 accuracy).

7. CONCLUSION

In this paper, we conducted a comprehensive analysis of *fake Likers* on Facebook collected from Fiverr and Microworkers by using the linkage and honeypot approaches. We compared how *fake Likers* from two different sources were different. In the comparison between *fake Likers* and legitimate *Likers*, we found that *fake Likers* were different from legitimate *Likers* with respect to liking behaviors, longevity, etc. Based on the analysis, we proposed 4 types of feature sets toward building accurate classification models. Experimental results show that our models significantly outperformed the baseline methods (i.e., PCA, SynchroTrap and CopyCatch) with accuracy = 0.871, false positive rate = 0.1, and false negative rate = 0.14. Under individual and simulated attack models, our approach consistently achieved over 0.82 of accuracy.

Acknowledgment. This work was supported in part by NSF grants CNS-1422215, CNS-1553035, and IUSE-1525601. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsors.

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] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
[3] Q. Cao, X. Yang, J. Yu, and C. Palow. Uncovering large groups of active malicious accounts in online social networks. In *CCS*, 2014.
[4] T. Chen and T. He. xgboost: extreme gradient boosting. 2015.
[5] E. De Cristofaro, A. Friedman, G. Jourjon, M. A. Kaafar, and M. Z. Shafiq. Paying for likes?: Understanding facebook like fraud using honeypots. In *IMC*, 2014.
[6] Facebook. SEC filings form 10-Q. http://tinyurl.com/zgnx9b6, 2016.
[7] J. Friedman, T. Hastie, R. Tibshirani, et al. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*, 28(2):337–407, 2000.
[8] J. Y. Jang, K. Han, P. C. Shih, and D. Lee. "Generation Like: Comparative characteristics in instagram. In *CHI*, 2015.
[9] K. Lee, B. D. Eoff, and J. Caverlee. Seven Months with the Devils: A Long-Term Study of Content Polluters on Twitter. In *ICWSM*, 2011.
[10] K. Lee, P. Tamilarasan, and J. Caverlee. Crowdturfers, campaigns, and social media: Tracking and revealing crowdsourced manipulation of social media. In *ICWSM*, 2013.
[11] K. Lee, S. Webb, and H. Ge. The dark side of micro-task marketplaces: Characterizing fiverr and automatically detecting crowdturfing. *ICWSM*, 2014.
[12] Y. Liu, Y. Liu, M. Zhang, and M. Shaoping. Pay me and i'll follow you: Detection of crowdturfing following activities in microblog environment. In *IJCAI*, 2016.
[13] M. Motoyama, D. McCoy, K. Levchenko, S. Savage, and G. M. Voelker. Dirty jobs: The role of freelance labor in web service abuse. In *USENIX Conference on Security*, 2011.
[14] G. Stringhini, M. Egele, C. Kruegel, and G. Vigna. Poultry markets: On the underground economy of twitter followers. In *Workshop on Online Social Networks*, 2012.
[15] G. Stringhini, C. Kruegel, and G. Vigna. Detecting spammers on social networks. In *ACSAC*, 2010.
[16] 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 Conference on Security*, 2013.
[17] 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 Conference on Security*, 2014.
[18] G. Wang, C. Wilson, X. Zhao, Y. Zhu, M. Mohanlal, H. Zheng, and B. Y. Zhao. Serf and turf: crowdturfing for fun and profit. In *WWW*, 2012.
[19] I. Wolfram Research. L2 norm. http://mathworld.wolfram.com/L2-Norm.html, 2016.
[20] Y. Yang and J. O. Pedersen. A comparative study on feature selection in text categorization. In *ICML*, 1997.