Instagram Fake and Automated Account Detection

Fatih Cagatay Akyon *, M. Esat Kalfaoglu †

*Elektrical & Electronics Engineering, Ihsan Dogramaci Bilkent University, Ankara, Turkey
†Electrical & Electronics Engineering, Middle East Technical University, Ankara, Turkey
{akyon}@ee.bilkent.edu.tr, {esatkalfaoglu}@gmail.com

***Abstract*—Fake engagement is one of the significant problems in Online Social Networks (OSNs) which is used to increase the popularity of an account in an inorganic manner. The detection of fake engagement is crucial because it leads to loss of money for businesses, wrong audience targeting in advertising, wrong product predictions systems, and unhealthy social network environment. This study is related with the detection of fake and automated accounts which leads to fake engagement on Instagram. As far as we know, there is no publicly available dataset for fake and automated accounts. For this purpose, two dataset have been generated for the detection of fake and automated accounts. For the detection of these accounts, machine learning algorithms like Naive Bayes, logistic regression, support vector machines and neural networks are applied. Additionally, for the detection of automated accounts, cost sensitive genetic algorithm is applied because of the unnatural bias in the dataset. To deal with the unevenness problem in the fake dataset, Smote-nc algorithm is implemented. For the automated and fake account detection problem, 86% and 96% are obtained, respectively.**

***Keywords*—fake engagement, machine learning, online social networks, Instagram, genetic algorithm, smote**

I. Introduction

Online Social Networks(OSNs) like Facebook and Instagram have becoming more and more popular and become the crucial part in Today's World. Beside the usage of OSNs as a medium of communication, they are also used to gain popularity and promote businesses. At the first glance, the popularity of an account is measured by some metrics like follower count or the properties of the shared contents like the number of likes, comments or views. Therefore, users of any social platforms might have a tendency to bolster its metrics in an artificial manner to get more benefits from OSNs.

There are some common ways to increase the reputation of an account in social media. These ways can be listed as usage of bots, buying social metrics such as like, comment and follower, and usage of some platforms or networks which enables users to trade metrics [1]. A bot is a piece of software that completes automated tasks over the Internet. By a 2018 study done by Ghost Data, nearly 95 million Instagram accounts are automated [2]. In 2016, bots generated more Internet traffic than humans [3]. Additionally, by the creation of fake accounts, vendors sells likes and followers very easily. For example, a company called IDigic sells 50k followers for only 250 dollars [4].

All of these actions listed above are inorganic and termed as fake engagement. In other words, fake engagement term covers all types of automated activities such as liking and commenting on posts, following accounts, uploading posts/stories. In addition, buying social media metrics can also be included in fake engagement terminology. The detection of users who inorganically grow its account is significant because this makes businesses pay more to users than its worth for advertising, makes advertisers reach to wrong audiences, make recommendation systems work inefficiently, make access to quality services and product harder.

978-1-7281-2868-9/19/$ 31.00

Fake engagement are divided into 2 separate topics which are the detection of automated accounts or bot accounts and fake accounts. As explained before, bot accounts are the users who performs automated activities like following users and liking media from related audience to increase its popularity metrics. Fake accounts are the accounts which are used to boost the social media metrics of a specific account who pays for this service. To highlight it more clearly, it can be also mentioned as fake followers. The main difference of automated and fake accounts is that automated account improves the metrics of itself while fake accounts improves the metrics of other users and creates unhealthy social media environment.

In the literature, there are some works and released datasets about the detection of fake engagement activity itself and the detection of users who engages inorganic activity in OSNs like Facebook and Twitter. The detection of Twitter fake accounts are studied in [3] using support vector machines and logistic regression, in [5] using graph based methods, in [6] using the joint usage of naive Bayes classifier with entropy minimization discretization. In [7], fake account detection on Twitter is studied by applying the GAIN measure [8] for weighting the all features used in the literature for this task and the improvements of such weigting on machine learning algorithms are shown. [9] is also on the detection of fake accounts using NLP and machine learning tools but also proposes some security architectures and focuces also on Facebook. In [10], the main focus is on the detection of fake followers on Twitter with some machine learning algorithms. In [11], fake social engagement on Youtube is studied by a graph diffusion process via local spectral subspace.

There are also some works in which Instagram are studied from the fake engagement perspective. Instagram has become one of the top social platforms. Instagram has reached about 1 billion monthly active user and 2 million monthly advertisers and users like 4.2 billions posts daily [12]. Therefore, it is crucial to preserve the healthy environment in such an important social platform. In [1], fake likes are tried to be determined on Instagram. In this study, the main concern is to estimate what is the probability that a user can like the

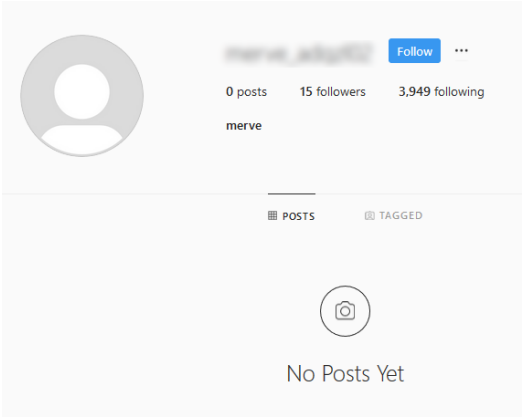

Figure 1: Fake account example from dataset. Most suspicious accounts are labeled as fake by hand.

Figure 2: In-class data distributions for "follower count" feature.

post of another user based on the network closeness, interest overlap, liking frequency, influencer effect and link farming hashtag effect. [13] and [14] are about the detection of spammy posts and spammy comments on Instagram, respectively. In [15], Facebook employees studied the detection of malicious accounts from the requests sent on Facebook and Instagram but these method is not applicable with publicly available information because requests are reachable only from Facebook. From all of these studies, it is observed that there is not any work done regarding fake and automated account detection for Instagram with publicly available information; moreover, no publicly available dataset is present required for these analysis. In this work, we collect and annotate fake account and automated account datasets and present a detailed analysis on fake and automated account detection for Instagram using machine learning algorithms and explain the steps required for preprocessing. The dataset is available on https://github.com/fcakyon/instafake-dataset.

For the rest of the paper, in Section II, fake account detection dataset and features are detailed. In Section III, automated account detection dataset and features and cost sensitive feature selection algorithm is given. In Section IV, implemented classification algorithms are detailed. Section V presents the results and section VI concludes the paper.

II. Fake Account Detection

This section is related with the detection of fake accounts. Fake accounts are the accounts which are used to increase the popularity metrics of other users. For this reason, they have a tendency to have a high following and low follower counts. Their liking behavior may look randomly. The absence of profile picture and strange user names are the common characteristics of fake accounts.

In this section, the dataset and the selected features for the dataset have been introduced for the detection of fake account. Then, the oversampling method is explained which is necessary for the unevenness in the number of real and fake account in the dataset.

A. Dataset and Features

For the dataset, 1002 real account and 201 fake account has been gathered after extensive manual labeling, including accounts from different countries and fields. During this gathering process, the points that is paid attention are follower and following counts, media counts, media posting dates or frequency, comments on media, some of the followed and following account, the existence of profile picture and the username of the profile.

An example fake profile from the dataset can be seen in Figure 1. As seen, it has a high following number of 3949, and low follower number of 15, has no profile picture and no posted media.

In the dataset, the selected base features can be listed as below:

- Total media number of the account.
- Follower count of the account.
- Following count of the account.
- Number of digits present in account username.
- Whether account is private, or not (binary feature).

To emphasize, all the features are not related with the user media, therefore the algorithm is not affected by the account privacy. The reason to add number of digits present in account username is that during the generation of fake accounts, some accounts are produced by adding different numbers to the same name. The number of digits distribution can be seen in Table II. As seen, more that 50% of fake accounts have more than one digits while real account has no digits with about 89%.

B. Oversampling

Distribution of classes in the fake account dataset is not even. This results in poor performance for the outnumbered class. SMOTE oversampling technique [16] is utilized to increase number of samples for fake accounts. K is chosen as 5 for this work. In the implementation of SMOTE, SMOTE-NC is applied which considers not only the quantity classes but

Figure 3: In-class data distributions for "following count" feature.

TABLE I: Distribution of accounts with changing number of digits included in their usernames.

# of digits	Real accounts	Fake account
0	88.9%	46.8%
1	2.5%	10.0%
2	5.3%	13.9%
3	0.7%	11.4%
3+	2.6%	17.9%

also the the categorical classes. After applying oversampling, all classifiers are trained on equal number of training samples per class (1002 per class).

III. Automated Account Detection

This section is related with the detection of automated accounts. Automated accounts which are also known as bot are the accounts which performs automated activities such as following, liking and commenting by targeting specific hashtags, locations of followers of specific accounts to increase their popularity metrics. Automated accounts might show fully inorganic behavior or organic and inorganic behavior together. The reason to observe organic behavior from such an account derives from the fact that user may continue to follow its own interests while the bot is running in background.

In this section, three subsection are presented which are Dataset and Features, Bias Problem and the Cost Sensitive Feature Selection. Because of the bias problem in the generated dataset, generic algorithm is applied with the weighting of selected features which is detailed in Cost Sensitive Feature Selection.

A. Dataset and Features

The dataset consists of 700 real account and 700 automated account gathered from different countries and fields. For the collection of real account, we have selected the people we know from our circle of friends. To collect the automated accounts, we examined the source codes of the most popular open source Instagram bots that form the essential portion of

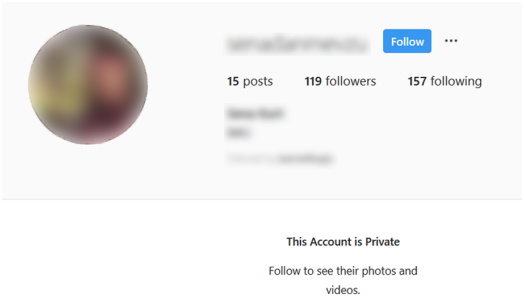

Figure 4: Example private account preview. Media details are not visible for private accounts.

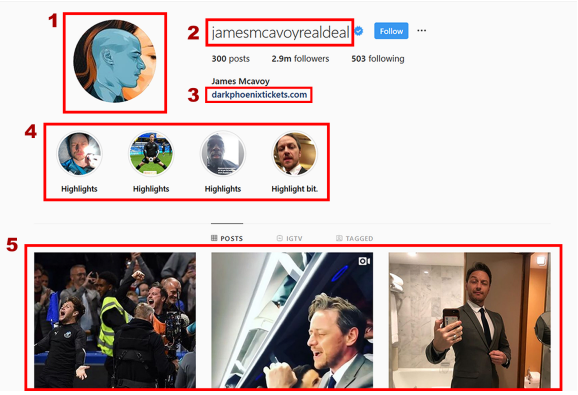

Figure 5: General preview of an Instagram profile. 1: Profile picture, 2: Username, 3: External URL, 4: Highlight reel, 5: User media

fake engagements and noted specific behaviors to tag these accounts. Hashtags are one of the most common ways for inorganic activity. For example, one of the criteria to detect fake likes on Instagram is to investigate like and follow trading hashtag usage [1]. In our case, most popular hashtags are targeted because it is observed that catching fake engagement is more easy and faster. From these hastags, if a user follows and unfollows after predefined exact durations (as observed from online instagram automations tool parameter sets), it is labelled as automated account. Instagram API is used with a Python wrapper to collect detailed media and user information of the accounts over 6 months period. For privacy reasons, any user related info is discarded for this work (user names, user photos, comment contents, hashtag names etc.).

Below are the scrapped base features from these accounts:

- Total media number of the accounts.
- Follower count of the account.
- Following count of the account.
- Whether account has at least one highlight reel, or not (binary feature).
- Whether account has external url in porfile, or not (binary feature).
- Number of photos user is tagged by someone else.

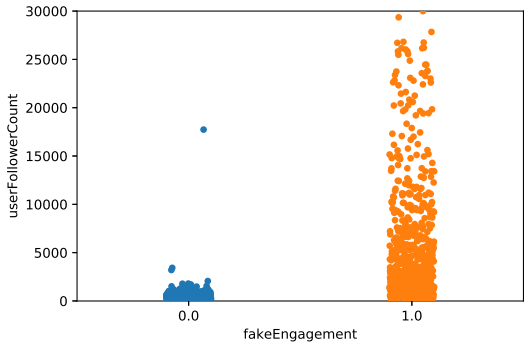

Figure 6: In-class data distributions for "follower count" feature.

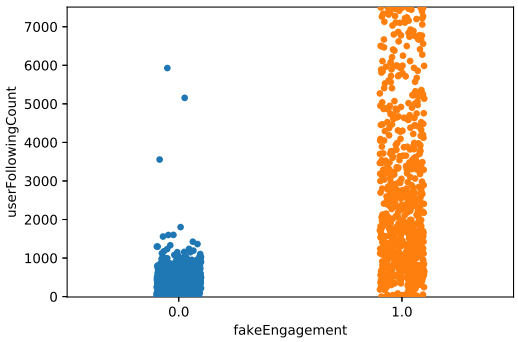

Figure 7: In-class data distributions for "following count" feature.

- Average recent media hashtag number.

If the account has no media, all features scrapped from user posts are assigned as 0. Furthermore, additional helpful features are derived using the base features such as:

- Average recent media like to comment ratio (LCR)
- Follower to following ratio (FFR).
- Whether account has not any media, or not (binary feature).

Here, "recent" means, the corresponding features are scrapped/calculated using only the media information that is posted in the last 18 months. To understand some of the features like highlight reel or private profile, Figure 4 and Figure 5 can be examined. To emphasize, the proposed automated account detection necessitates the access to user media.

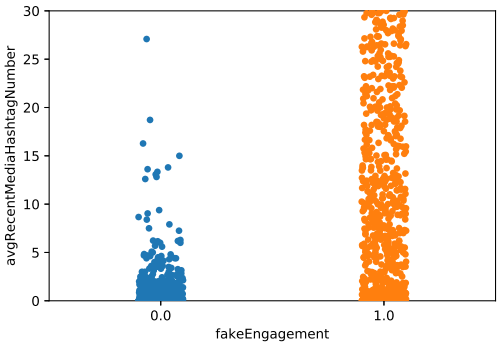

Figure 8: In-class data distributions for "average hashtag number" feature.

B. Bias Problem

There was some negative (unrealistic) bias present in some of the features. In Figures 6-8, in-class distribution of the whole dataset with respect to chosen continuous features are illustrated. "Fake Engagement" being 1 correspond to the accounts involved in fake engagement or automated account while 0 correspond to the accounts that only has natural engagements or real account. As seen from the figures, chosen features have bias over the dataset. Although the bias in follower and following numbers is unrealistic (we have undeliberately chosen accounts with low follower&following numbers as real accounts, it does not reflect the real situation), bias in average hashtag number is nearly natural (accounts with fake engagement tend to use more hashtags per post).

In Tables II-III, projection of the dataset over chosen binary features are given. As can be seen from tables, there are also bias present over these features, however this time; these are realistic bias. Highlight reels can be considered as an effective separator while real engagement accounts mostly not having url present can also considered to be a true bias.

C. Cost Sensitive Feature Selection

To overcome these unrealistic biases and select the most effective features, a cost sensitive genetic feature selection algorithm is developed. Pseudo code for the this genetic algorithm can be seen in Algorithm 1. Firstly, normalization is applied for the continuous features while binary features remained the same. Then, the normalized data is given to the genetic algorithm for a cost sensitive feature selection. An individual, whose length is same as total feature number of the data, is an array consisting of 1's and 0's (depending on if the feature is selected or not). To illustrate, if the second element of the individual is 1, then it means the second feature is one of the selected features for this specific individual. Using this representation, randomly generated individuals are used to form a population.

$$Fitness = F2\ Score - 2 \times Tot.Feat.Cost \tag{1}$$

$$F2\ Score = 5 \times \frac{precision \times recall}{5 \times precision + recall} \tag{2}$$

Fitness calculation formula is given in Eq. 1. Here $Tot.Feat.Cost$ is calculated by summing the individual costs of the selected features. Feature costs can be seen in Table IV. These costs are determined based on the reliability of the data collection which is discussed in the previous Bias Problem section. Realistic biases are represented with lower features costs while the negative bias is represented with higher costs. For the F2 score in Eq. 1, it belongs the classifier loss. F2 score

TABLE II: In-class data distributions for "user has highligh reels" feature.

	Real Engagement	Fake engagement
Has Not Any Highligh Reels	468	260
Has Highligh Reels	232	440

TABLE III: In-class data distribution for "user has highligh reels" feature.

	Real Engagement	Fake engagement
Has Not Any External Url	654	317
Has External Url	46	383

formula is given in Eq. 2. For the calculation of F2 score, a two layer neural network architecture has been implemented which is detailed in Table V. Then, the selected features are used as an input to the all classification methods which will be detailed in the Classification Methods section.

As the genetic operations; elitism, randomness, tournament based crossover, and mutation operations are implemented. At each generation, the individual having best fitness is directly selected for the next generation.

Calculated fitness values of the fittest individual of the population for a given generation can be seen in Figure 9. As expected, evolution results in monotonic increase in the best fitness (considering mutation rate is very low). After 10 generations, individual with the best fitness value is used to select the best features. The selected features and total cost for selecting these features are given in Table IV.

IV. Classification Methods

Several traditional and neural network based learning methods are implemented as classifiers. As traditional methods, Naive Bayes, logistic regression and support vector machine (SVM) is employed. In Naïve Bayes method, independent features of different classes are exploited to form the posterior distributions of the classes, and maximum a-posteriori (MAP) estimation is performed. Logistic regression again exploits independent features to differentiate two classes. SVM focuses on finding a hyperplane which separates a dataset in the best way. In addition to preprocessed data(features), raw data can also be used as inputs while training and testing these type of networks.

V. Results

Through utilization of different kinds of algorithms, it is aimed to exploit different aspects of dataset (i.e., independence, separability, complex relations) which has not been deeply considered in literature and to find a good way of detection of the fake and automated accounts of Instagram.

For the detection of automated accounts, cost sensitive selected features given in Table VI have been used. For the detection of fake accounts, the base features of the fake-real dataset has been used directly.

TABLE IV: Costs of the features based on their bias reliability.

Features	Cost
Total media number	2
Follower count	4
Following count	4
Has highlight reel	2
Has external url	2
Tag number	3
Average hashtag number	2
Has 0 media	1
LCR	2
FFR	4

TABLE V: Neural Network Details.

# of Layers:	2
# of Hidden Units (per layer):	32
Optimization:	$ADAM\ with\ Minibatch$
Non-linearity:	$ReLu$
Loss Function:	$Categorical\ Crossentropy$
Learning Rate:	0.001
Minibatch Size:	64
Epochs:	100
Train-Test Split:	$\%70 - \%30$

For the detection of automated accounts, to compare and test the effectiveness of the implemented techniques; Precision, Recall and F1 Score are used as the evaluation metric as given in equations 3-5 respectively. Terms TP, TN FP, and FN present in these equations correspond to True Positive, True Negative, False Positive and False Negative given in 10. F1 Score is more meaningful for performance evaluation because precision ignores the effect of FN and recall ignores the effect of FP. F1 score considers both of them.

For a fair comparison, parameter optimization is performed by grid search for the classifiers that rely on parameters. Extensive 10-fold cross validations performed over the training portion of the dataset. Kernel is chosen as $radial\ basis\ function$, the $gamma$ parameter (kernel coefficient) is chosen as 1, and the penalty parameter C is chosen as 100 (mid level regularization) for the SVM. $Solver$ is chosen as $Newton\ Conjugate\ Gradient$, $inverse\ of\ regularization\ coefficient$ (smaller value corresponds to stronger regularization) is chosen as 1000 (tighter regularization), and $convergence\ tolerance\ level$ is chosen as 0.1 for the logistic regression technique. Neural network is run with the parameters given in Table V.

Test results for fake account detection dataset can be found in Table VII. As mentioned before, SMOTE-NC has been used for the oversampling. F1 scores are calculated by the macro average method. The reason to use macro average is the fact that the distribution in the data does not reflect the real distribution in the Instagram. It is desired to place importance equally on fake and real users. From the table, it is observed that oversampling has increased the performance of all methods. The highest performance without oversampling

Algorithm 1 Genetic Algorithm

Inputs: **FullDataset**: *(Normalized) Dataset containing all of the features*, **PopulationSize**: *Number of individuals present at each generation*, **NumberofGenerations**: *Number of generations to be iterated on*, **MutationRate**: *Mutation probability*
Outputs: **ReducedDataset**: *Dataset containing only the selected features*
Initializations :Initalize **Population** randomly
for $ind = 1$: **NumberofGenerations do**
  Calculate **Population** fitness
  Select the best individual (elitisim)
  Select 1 random individual (randomness)
  Perform xover to rest of the individuals (tournament)
  Mutate 1 individual with prob. **MutationRate**
  Update **Population**
end for
Form **ReducedDataset** using **Population**
return ReducedDataset

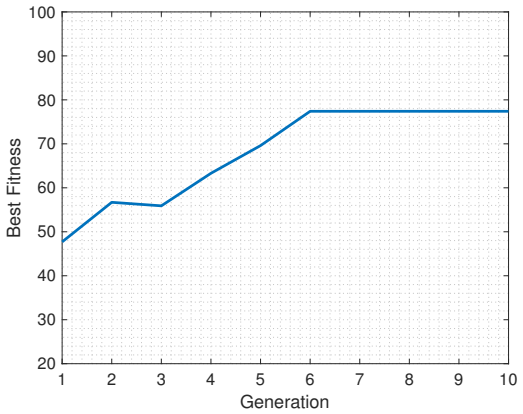

Figure 9: Fitness of the fittest individual of each generation.

is observed with the neural network, while SVM and neural network perform equally in oversampling case with 94%.

Test results for automated account detection dataset can be found in Table VIII. Neural network and SVM has the best overall F-1 scores. It is expected since it is known that neural networks can learn complex mappings with enough training data and SVMs are well known for optimizing the margin better than most other algorithms in binary tasks. Poor performance from Naive Bayes and logistic regression was not surprising since the features are not distinctly independent (considering these methods highly rely on the independence of the features). Moreover, low precision present for the Naive Bayes with Gaussian distribution means that the true distribution of the labels does not corporate with this likelihood assumption.

		Predicted class	
Actual Class		Class = Yes	Class = No
	Class = Yes	True Positive	False Negative
	Class = No	False Positive	True Negative

Figure 10: Actual class vs predicted class comparison.

TABLE VI: Corresponding features of the fittest individual and their cost.

Selected Features	Cost
Total media number	2
Has external url	2
Average hashtag number	2
LCR	1
Total	**7**

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

$$\text{F1 Score} = 2 \times \frac{Recall \times Precision}{Recall + Precision} \tag{5}$$

TABLE VII: Evaluation of the classifiers over the fake account dataset.

Classifier	F1 score without oversampling	F1 score with oversampling
Support Vector Machine	88.2%	**94.0%**
Naive Bayes (Bernoulli Dist.)	83.8%	88.2%
Naive Bayes (Gaussian Dist.)	54.2%	65.6%
Logistic Regression	87.8%	90.8%
Neural Network	**89.0%**	**94.0%**

TABLE VIII: Evaluation of the classifiers over the fake engagement dataset.

Classifier	Precision	Recall	F1-Score
Support Vector Machine	**91%**	82%	**86%**
Naive Bayes (Bernoulli Dist.)	85%	68%	78%
Naive Bayes (Gaussian Dist.)	51%	**98%**	67%
Logistic Regression	80%	70%	75%
Neural Network	89%	84%	**86%**

VI. Conclusions

In conclusion, detection of the fake and automated accounts which leads to fake engagement in Instagram is studied as a binary classification problem in this paper. To our knowledge, this is the first time for such an analysis over Instagram accounts. Our contributions with this work are: collection of datasets for fake and automated account detection, proposing derived features for fake and automated classification, proposing a cost sensitive feature reduction technique based on genetic algorithms for selecting best features for the classification of automated accounts, correcting the unevenness in the fake account dataset using the SMOTE-NC algorithm and evaluating several pattern recognition methods over the collected datasets. As a result, SVM and neural network

based methods achieved the most promising F1 score for the detection of automated accounts with 86% and neural network achieved the best F1 score performance with 95%.

As a future work, recurrent neural networks can be utilized for the time series user data for a better detection of automated accounts. The biased features in the automated account dataset can be balanced by finding the suitable real users. Fake user detector explained in this paper can also be used for finding the suitable real users in the automated account dataset.

REFERENCES

[1] I. Sen, A. Aggarwal, S. Mian, S. Singh, P. Kumaraguru, and A. Datta, "Worth its weight in likes: Towards detecting fake likes on instagram." in *WebSci*, 2018, pp. 205–209.

[2] T. Information, "Instagram's Growing Bot Problem," www.theinformation.com/articles/instagrams-growing-bot-problem, accessed: 2019-06-10.

[3] P. G. Efthimion, S. Payne, and N. Proferes, "Supervised machine learning bot detection techniques to identify social twitter bots," *SMU Data Science Review*, vol. 1, no. 2, p. 5, 2018.

[4] "Influencer fraud," *Influencer Marketing Hub*, 2018.

[5] M. Mohammadrezaei, M. E. Shiri, and A. M. Rahmani, "Identifying fake accounts on social networks based on graph analysis and classification algorithms," *Security and Communication Networks*, vol. 2018, 2018.

[6] B. Erşahin, Ö. Aktaş, D. Kılınç, and C. Akyol, "Twitter fake account detection," in *2017 International Conference on Computer Science and Engineering (UBMK)*. IEEE, 2017, pp. 388–392.

[7] A. El Azab, A. M. Idrees, M. A. Mahmoud, and H. Hefny, "Fake account detection in twitter based on minimum weighted feature set," *Int. Sch. Sci. Res. Innov*, vol. 10, no. 1, pp. 13–18, 2016.

[8] A. G. Karegowda, A. S. Manjunath, and M. A. Jayaram, "Comparative study of attribute selection using gain ratio and correlation based feature selection," 2010.

[9] R. Raturi, "Machine learning implementation for identifying fake accounts in social network," *International Journal of Pure and Applied Mathematics*, vol. 118, no. 20, pp. 4785–4797, 2018.

[10] S. Cresci, R. Di Pietro, M. Petrocchi, A. Spognardi, and M. Tesconi, "Fame for sale: Efficient detection of fake twitter followers," *Decision Support Systems*, vol. 80, pp. 56–71, 2015.

[11] Y. Li, O. Martinez, X. Chen, Y. Li, and J. E. Hopcroft, "In a world that counts: Clustering and detecting fake social engagement at scale," in *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2016, pp. 111–120.

[12] T. Clarke, "22+ Instagram Stats That Marketers Can't Ignore This Year," https://blog.hootsuite.com/instagram-statistics/, accessed: 2019-06-25.

[13] W. Zhang and H. Sun, "Instagram spam detection," in *2017 IEEE 22nd Pacific Rim International Symposium on Dependable Computing (PRDC)*, Jan 2017, pp. 227–228.

[14] A. Akbar Septiandri and O. Wibisono, "Detecting spam comments on indonesia's instagram posts," *Journal of Physics: Conference Series*, vol. 801, p. 012069, 01 2017.

[15] Q. Cao, X. Yang, J. Yu, and C. Palow, "Uncovering large groups of active malicious accounts in online social networks," *Proceedings of the ACM Conference on Computer and Communications Security*, pp. 477–488, 11 2014.

[16] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.