Instagram Fake and Automated Account Detection

Ansh Gupta (2020A7PS0116P)

Nikhil Pradhan (2020A7PS1205P)

Mithil Shah (2020A7PS0980P)

Birla Institute Of Technology And Science, Pilani

CS F415: Data Mining

Dr. Pratik Narang

Prof. Yashvardhan Sharma

14th April, 2023

Abstract

Fake engagement on Online Social Networks (OSNs) using automated accounts or fake accounts is an emerging problem which increases the popularity of an account in an inorganic manner. This inorganic growth 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 and creates an unhealthy social network environment. This study aims to tackle this problem for a popular networking app Instagram. For the detection of fake accounts, machine learning algorithms like Naive Bayes, Logistic Regression, Support Vector Machines and Neural Networks are applied. Additionally, for the detection of automated accounts, a cost sensitive genetic algorithm is proposed to handle 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 datasets, 86% and 96% classification accuracies are obtained, respectively.

Keywords: fake engagement, inorganic growth, machine learning, online social networks (OSN), Instagram, genetic algorithm, smote.

Instagram Fake and Automated Account Detection

What work are you proposing for the term project?

The term project proposes detection of fake and automated accounts, which lead to fake engagement in Instagram. We will be taking a research paper by Fatih Cagatay Akyon and Esat Kalfaoglu on the topic Instagram Fake and Automated Account Detection [1] as our term project. We are using models such as Naive Bayes , SVM and neural networks to classify such accounts. We aim at reproducing the results and inferences mentioned in the above paper and achieve better accuracy in fake and automated accounts detection.

Why is the task/project interesting and important?

The project is interesting and important because it addresses the problem of fake engagement on Instagram, which is a significant issue in today’s world. This project is crucial as it leads to loss of money in businesses, wrong audience getting targeted which could lead to building wrong recommendation system models.

What is the challenge in it?

The challenge in such papers is generating datasets for models to be trained on, since there are very few known datasets on fake accounts, it becomes difficult to generate a model. There also exists the problem of lack of evenness between the real and fake datasets, since the real dataset will far exceed the count of the fake dataset.

What is the prior work in this space? Describe in brief.

There has been some prior research in the detection of fake engagement activity and users engaging in inorganic activity on social media platforms like Twitter, Facebook, and Instagram.

For Twitter, several studies have used machine learning algorithms such as support vector machines, logistic regression [2], graph-based methods [3], naive Bayes classifiers, and entropy minimization discretization to detect fake accounts and fake followers. One study used the GAIN measure to weigh all features used in the literature and improve machine learning algorithms [4].

For YouTube, one study used a graph diffusion process via a local spectral subspace to detect fake social engagement [5].

For Instagram, studies focused on detecting fake likes, spammy posts, and spam comments using network closeness [6–7], interest overlap, liking frequency, the influencer effect, and the link farming hashtag effect [8].

Overall, the prior work in this area has used various machine learning algorithms and techniques to identify fake engagement activity and users engaging in inorganic activity on social media platforms. Each of the papers in this space uses machine learning along with some kind of preprocessing to train the model.

What is the approach taken by you and how is it novel/interesting in the context of the prior work?

The author of the paper “Instagram Fake and Automated Account Detection” has created datasets for the fake accounts as well as the automated accounts. Scraping techniques helped to achieve this. Some of the features extracted were then used to create derived features, which were then given to the model. The author has also implemented cost-sensitive feature reduction techniques based on genetic algorithms for selecting the best features for the classification of the model.

Describe a high-level idea of how your method will work.

Using the dataset given, we would first generate a few derived features such as the average recent media like to comment ratio (LCR), the follower to following ratio (FFR), and whether the account has any media or not. We then ran on oversimplification algorithm for the fake/real dataset to make the classes balanced. Next, we ran a cost-sensitive feature selection algorithm an the Automated/Non-automated dataset to remove the negative bias present in the database. For classification, we will be used SVM, a 3 layer Neural Network, Bernoulli Naive Bayesian Classifier and Gaussian Naive Bayesian Classifier.

What all datasets will you use? Provide links, year of release, size, etc.

The dataset is publicly available on https://github.com/fcakyon/instafake-dataset . This dataset was released to the public in 2019 by the authors of the paper [1]. For fake accounts detection it contains data of 1002 real accounts and 201 fake accounts data, ranging from different countries and fields. For automated accounts detection it consists of 700 real accounts and 700 automated accounts data, ranging from different countries and fields.

What data mining (preprocessing / machine learning / classification) approaches will be used in your work?

1. Oversampling:

   Since the original dataset for fake/real accounts is very imbalanced, we have oversampled the dataset using the SMOTE-NC algorithm. Sometimes, classification datasets have an imbalance in the representation of classes. Models trained on such datasets tend to ignore and consequently have a poorer performance on the minority class. One method to tackle this issue is oversampling i.e. adding more entries of the class with fewer number of objects present. SMOTE (Synthetic Minority Oversampling Technique) is an oversampling algorithm that duplicates values of the minority class. Since, SMOTE only works on a set of continuous attributes, we have used SMOTE-NC (NC stands for nominal-continuous) which is used to oversample datasets with both continuous and categorical features. The dataset initially contained 994 real and 200 fake accounts. After using SMOTE-NC, we have 994 real and 994 fake accounts.

2. Cost sensitive feature selection using genetic algorithm

   Before training the model to predict fake and automated accounts, we have selected features that will be relevant to the model using a genetic algorithm which uses feature costs. In this model, the chromosome is a list of 0s and 1s, whose length equals the number of columns in the dataset. 0 in the ith index means that the ith column is excluded in the selection and vice versa. The fitness function we are using for this genetic algorithm is:

   $$\text{Fitness} = \text{F2 score (in \%)} - (2 * \text{Total Feature Costs})$$

   Where,

   $$\text{F2 score} = (\text{precision} * \text{recall}) / (\text{precision} + \text{recall})$$

The total feature cost is the sum of individual feature costs of all features included in the selection. These individual costs were assigned based on their bias values.

The F2 score is calculated using a 2 layer neural network that has 32 hidden units, uses Adam optimization, ReLu as the activation function, categorical crossentropy as the loss function, 0.001 as the learning rate, with a minibath size of 64, 100 epochs and a train-test split of 70%-30%.

The genetic operations used in the feature selection algorithm are eliteness, randomness, crossover and mutation

Firstly, the individual with the best fitness and another random individual are crossed over using a tournament based crossover. Then a certain set of features are mutated based on the mutation rate. This new individual is then added to the population. This process is repeated 10 times and the final best individual is chosen as the feature set.

Since the feature costs mentioned in the paper only belonged to the automated/non-automated dataset, we have used the feature selection algorithm only for the automated/non-automated model.

3. Machine Learning

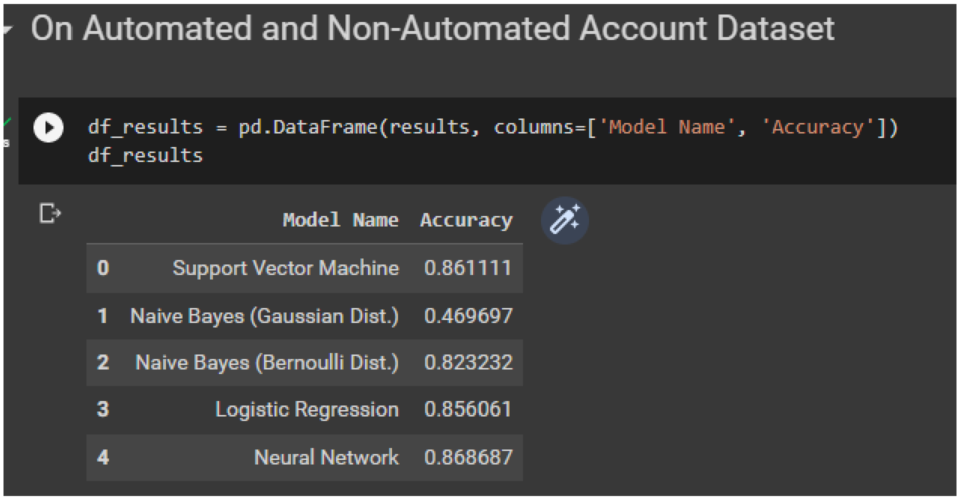
   We have implemented multiple machine learning models on both data sets with different levels of success. The models used include Support Vector Machine, Gaussian Naive Bayesian Classifier, Bernoulli Naive Bayesian Classifier, and a Neural Network. The neural network. The neural network uses 3 dense layers with 50, 150 and 25 nodes respectively. We have used ReLu and softmax as the non-liearity with a dropout of 0.2.

How will you evaluate the method, i.e. what performance metrics will you use, and what baselines will you compare to?

We have used accuracy to judge the correctness of the model

Accuracy and Findings

The accuracy of the models for the Automated / Non-Automated Classification without feature selection is as follows:

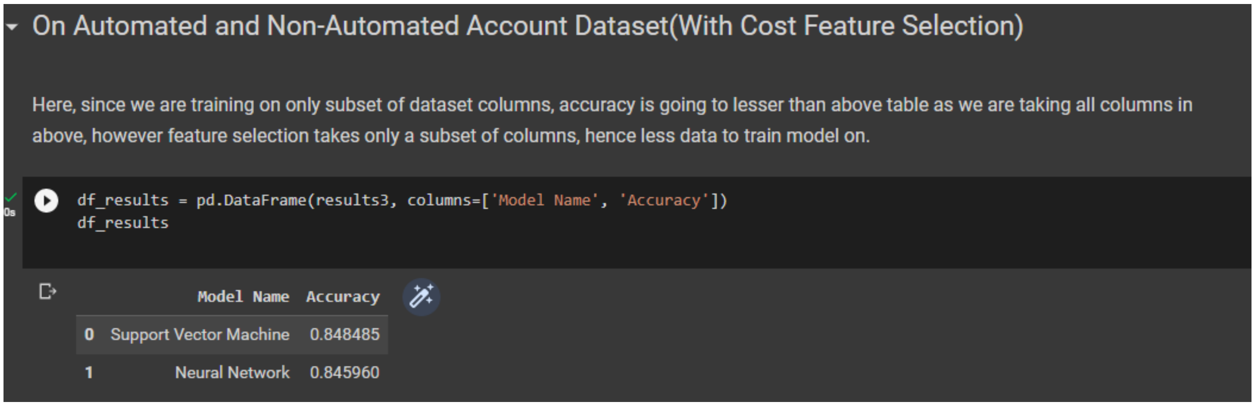


On Automated and Non-Automated Account Dataset

```
df_results = pd.DataFrame(results, columns=['Model Name', 'Accuracy'])
df_results
```

	Model Name	Accuracy
0	Support Vector Machine	0.861111
1	Naive Bayes (Gaussian Dist.)	0.469697
2	Naive Bayes (Bernoulli Dist.)	0.823232
3	Logistic Regression	0.856061
4	Neural Network	0.868687

The accuracy of the models for the Automated / Non-Automated Classification with feature selection is as follows:



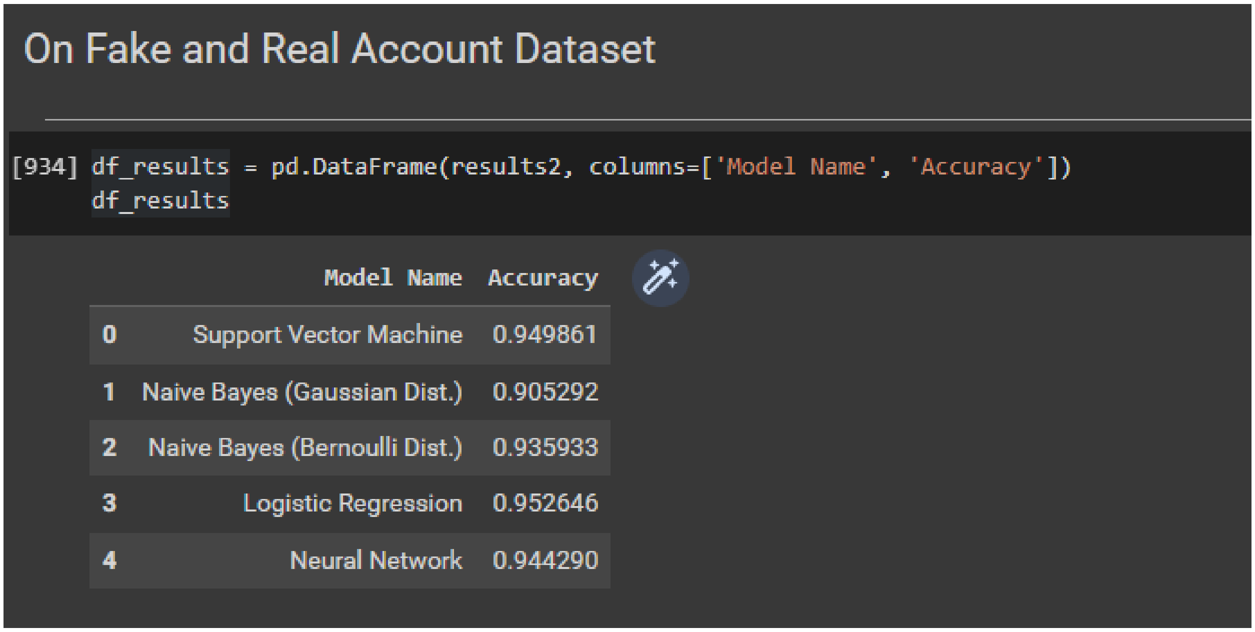
On Automated and Non-Automated Account Dataset(With Cost Feature Selection)

Here, since we are training on only subset of dataset columns, accuracy is going to lesser than above table as we are taking all columns in above, however feature selection takes only a subset of columns, hence less data to train model on.

```
df_results = pd.DataFrame(results3, columns=['Model Name', 'Accuracy'])
df_results
```

	Model Name	Accuracy
0	Support Vector Machine	0.848485
1	Neural Network	0.845960

The accuracy of the models for theFake/Real Classification without feature selection is as follows:



On Fake and Real Account Dataset

```
[934] df_results = pd.DataFrame(results2, columns=['Model Name', 'Accuracy'])
      df_results
```

	Model Name	Accuracy
0	Support Vector Machine	0.949861
1	Naive Bayes (Gaussian Dist.)	0.905292
2	Naive Bayes (Bernoulli Dist.)	0.935933
3	Logistic Regression	0.952646
4	Neural Network	0.944290

Result Analysis and Interpretation

The models have higher accuracy scores for the fake/real accounts dataset as compared to the automated/non-automated dataset. Neural networks and logistic regression were the 2 most successful models for both of these datasets. Although the models had a lower accuracy with cost sensitive feature selection, it does not necessarily mean that the genetic algorithm was incorrect since reducing the features gives the models less data to work with.

In our study, since the feature selection model trains on only subset of dataset columns, accuracy is going to be lower than if we took the whole dataset table as we are taking all columns in it, however, feature selection takes only a subset of columns, hence less data to train the model on, leading to lower accuracy.

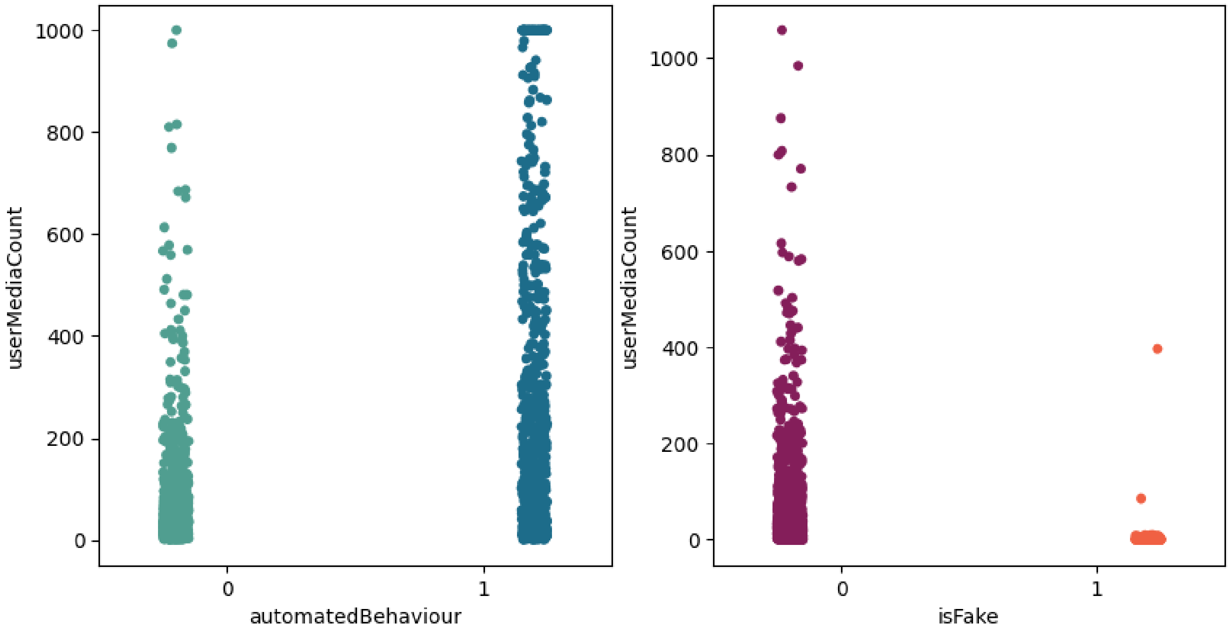


Figure 1. Real accounts and non-automated accounts post more media on their accounts than fake accounts whereas automated accounts tends to post the most among these

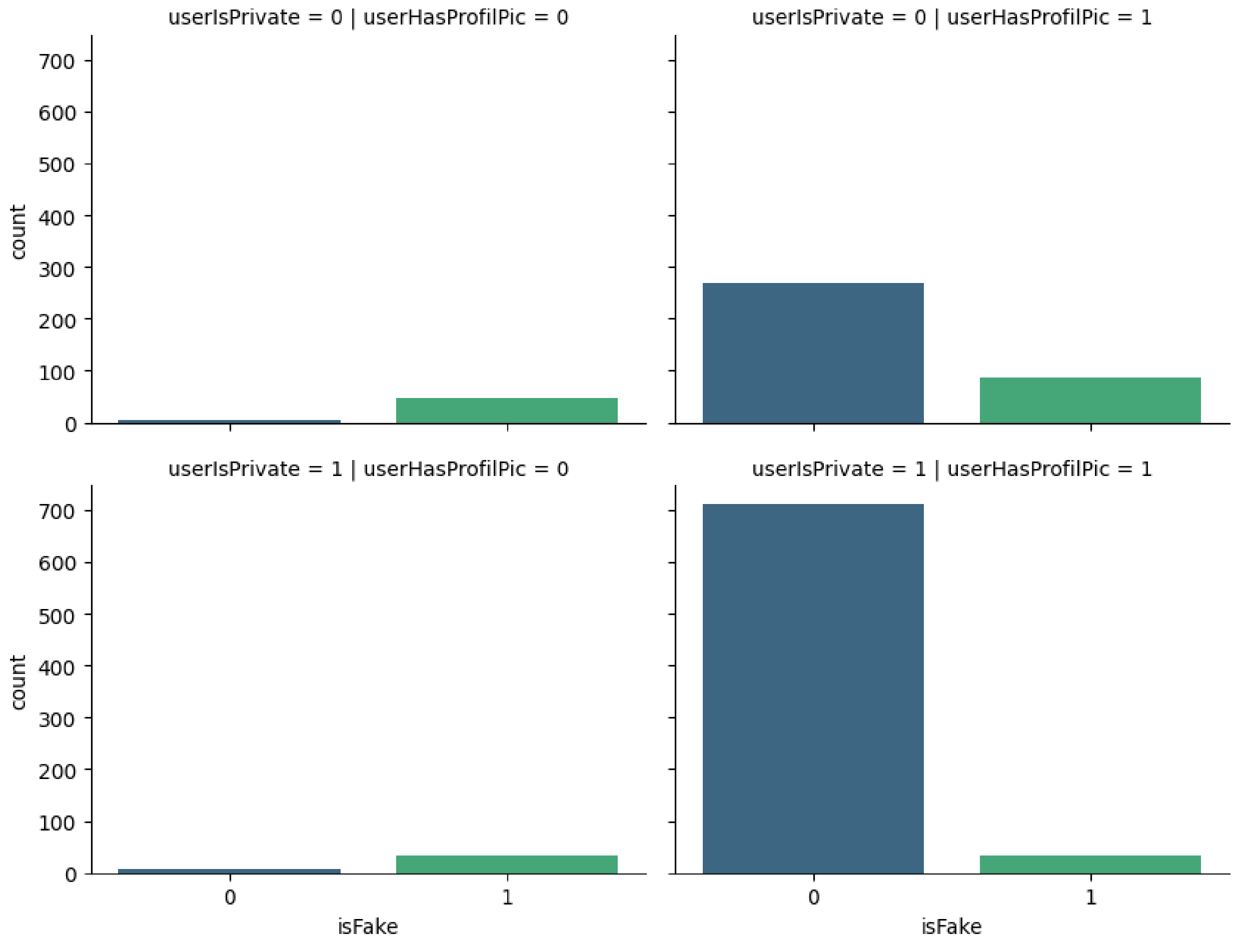


Figure 2. Private account having profile pic are more likely to be real

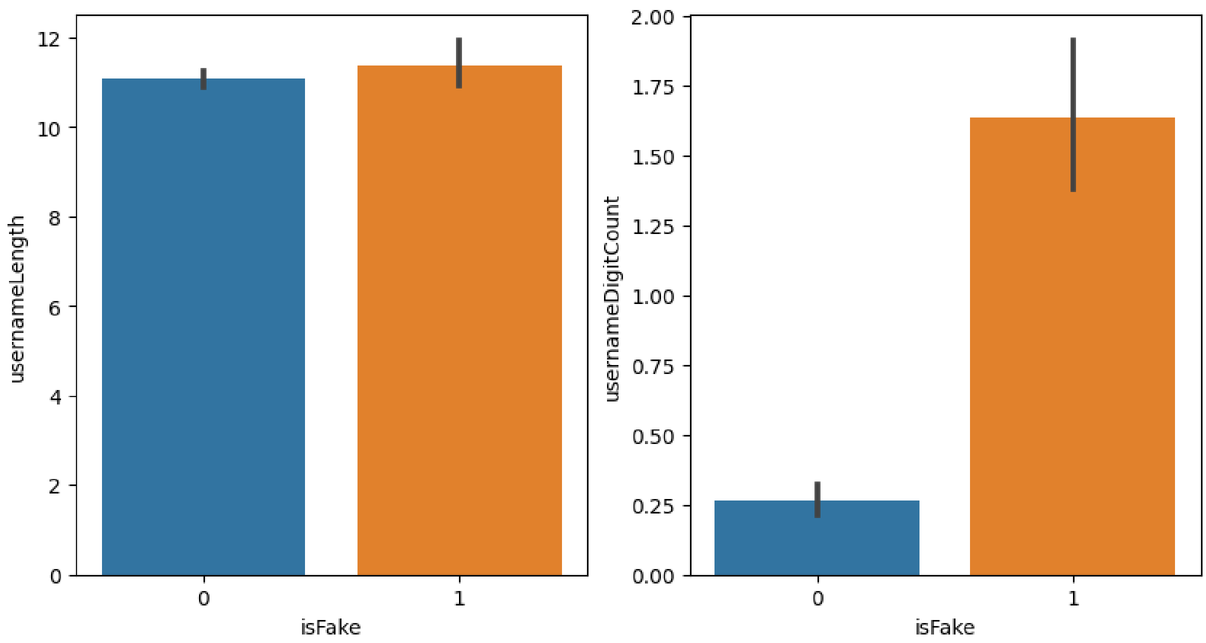


Figure 3. Accounts having username with more number of digits are found to be fake

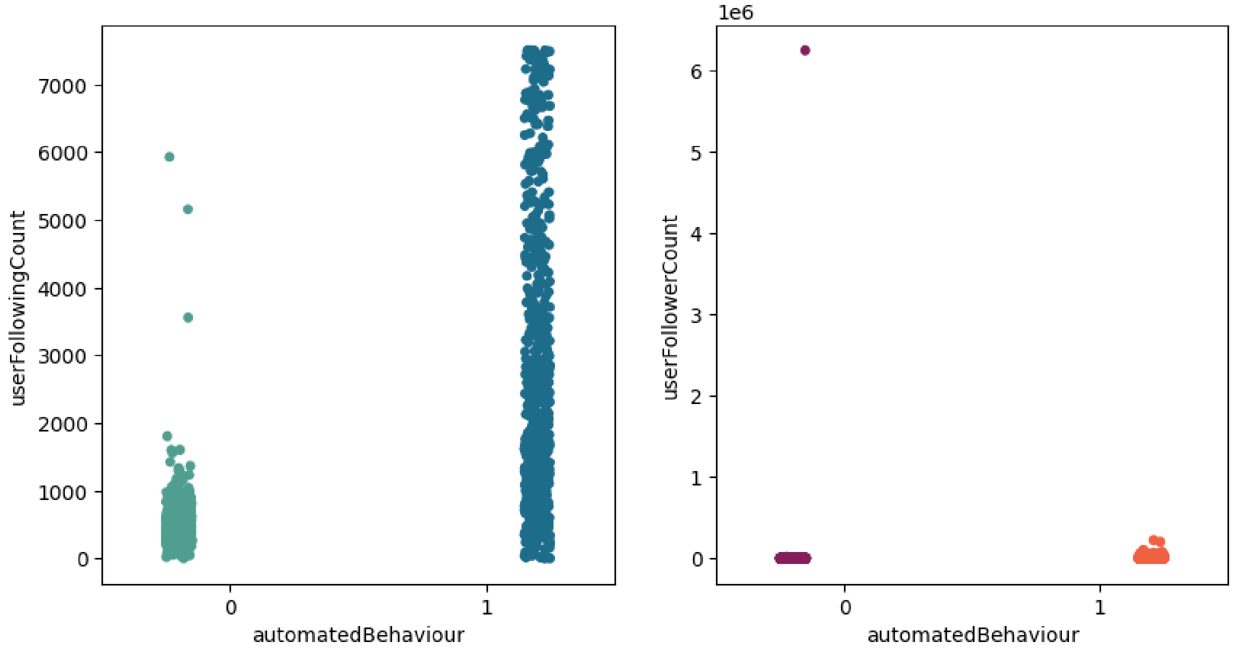


Figure 4. Automated accounts are found to have more user following

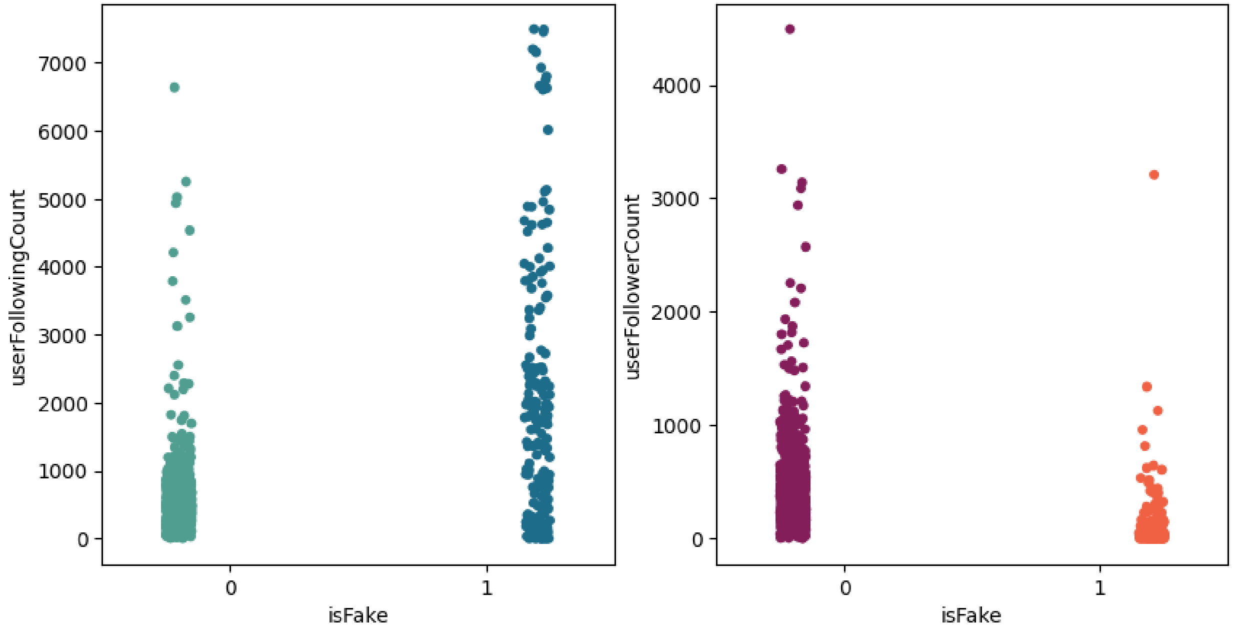


Figure 5. Accounts having more following count are more likely to be fake

References

[1] Akyön, Fatih Çağatay & Kalfaoglu, Esat. (2019). Instagram Fake and Automated Account Detection. 1-7. 10.1109/ASYU48272.2019.8946437.

[2] P. G. Efthimion, S. Payne, ve N. Proferes, “Supervised machine learning bot detection techniques to identify social twitter bots,” SMU Data Science Review, vol. 1, no. 2, p. 5, 2018.
[3] M. Mohammadrezaei, M. E. Shiri, ve A. M. Rahmani, “Identifying fake accounts on social networks based on graph analysis and classification algorithms,” Security and Communication Networks, vol. 2018, 2018
[4] A. G. Karegowda, A. S. Manjunath, ve M. A. Jayaram, “Comparative study of attribute selection using gain ratio and correlation based feature selection,” 2010
[5] Y. Li, O. Martinez, X. Chen, Y. Li, ve J. E. Hopcroft, “In a world that counts: Clustering and detecting fake social engagement at scale,” Proceedings of the 25th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee, 2016, sf. 111–120
[6] W. Zhang ve H. Sun, “Instagram spam detection,” 2017 IEEE 22nd Pacific Rim International Symposium on Dependable Computing (PRDC), Jan 2017, sf. 227–228.
[7] A. Akbar Septiandri ve O. Wibisono, “Detecting spam comments on Indonesia's instagram posts,” Journal of Physics: Conference Series, vol. 801, p. 012069, 01 2017.
[8] I. Sen, A. Aggarwal, S. Mian, S. Singh, P. Kumaraguru, ve A. Datta, “Worth its weight in likes: Towards detecting fake likes on instagram.” WebSci, 2018, sf. 205–209.