Methods for Detecting Fake Accounts on the Social Network VK

Alexey D. Frunze
National Research Nuclear University MEPHI
(Moscow Engineering Physics Institute)
Moscow, Russia
Frunze111alex@gmail.com

Aleksey A. Frolov
National Research Nuclear University MEPHI
(Moscow Engineering Physics Institute)
Moscow, Russia
aleksey2093@outlook.com

2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus) | 978-1-6654-0476-1/20/$31.00 ©2021 IEEE | DOI: 10.1109/ElConRus51938.2021.9396670

***Abstract*—This article addresses the issue of detecting fake user accounts on social networks. The purpose of the study is to find and analyse the distinctive features of fake accounts in comparison with the accounts of real users using the example of the social network VK. The introduction looks at various studies on this topic, including the possibility of identifying fake accounts on other social networks. The experimental part describes the process of collecting information about the accounts of fake users from the social network VK, highlighting features, compiling a dataset for training the classifier. The results show the results of the accuracy of detecting fake accounts and a comparison of the results of various classification methods. In the conclusion, conclusions are presented about the effectiveness of the proposed approach and possibilities.**

Keywords—Social media; fake; machine learning; VK

I. Introduction

Social media is widespread these days. They are used both for communication, learning, meeting new people [1], and for business and advertising [2]. One of the problems with social networks is fake accounts, which are used for various anonymous actions. Fake accounts can be used both for deception, blackmail and extortion, which adversely affects people's trust in each other on social networks, and for spreading fake news [3], recruiting into terrorist organizations and driving people to suicide [4]. However, it should be noted that fake accounts are used not only for harm, but also for various personal purposes that do not affect ordinary users, but such accounts interfere with researchers working in the field of social network analysis [5]. In this regard, it was decided, within the framework of this work, to conduct a study to identify fake on the social network VK.

II. Problem Statement

The study [6] examines methods of calculating fake accounts on the social network Instagram. The authors highlight key features of fake accounts and train various classifiers: random forest method, J48, support vector machine, RBF, MLP, naive Bayesian Classifier, Heffding tree, packed decision tree. Based on the training results, the packed decision tree became the most accurate method, its accuracy was 98.45%. MLP, the random forest method and the Heffding tree also performed well. All of them showed accuracy of more than 96%.

The authors of the study [7] proposed a method for detecting fake accounts on the social network Twitter. A naive Bayesian classifier is used to successfully compute fakes. The authors also used data preprocessing using the Entropy Minimization Discretization (EMD) method. For a successful calculation, the study identifies 16 different signs of fake accounts. The research results compare two classification results: with and without data preprocessing. The calculation accuracy without using EMD was 86.1%, and with this method 90.9%.

The authors of [8] conducted a study on the calculation of fake social media accounts using the Blacklist. The research was based on the social network Twitter. The authors analyze the text on the pages of fake accounts and based on the analysis, add words to the Blacklist. The accuracy of this method was 95.4%.

In the article [5], a study was carried out on the social network VK, within the framework of which fake accounts were calculated. In the work, the key features are identified, and the classifier is trained. The method used for the classification was the random forest method. Of the advantages of this method, it is noted that the algorithm is not sensitive to unnecessary variables and does not require data transformation. Of the minuses, the authors note the tendency of the method to retrain. According to the classification results, the calculation accuracy was 92%.

Also, several studies on calculating fake were conducted based on the social network Facebook.

In [9], 12 machine learning methods were analyzed. The methods that scored more than 75% accuracy were J48, the random forest method, the random tree method, the naive Bayesian classifier, and others.

In [10], the Facebook algorithm was considered, which calculates and blocks fake accounts. The study notes that the algorithm learns on the fly and is increasingly successful at blocking fake accounts.

III. Research Goal

The aim of this research is to find a method to successfully calculate fake accounts using machine learning methods. For this it was necessary:

1. Analyze the capabilities and features of the VK social network
2. To study the possibilities provided by the VK API [11]
3. Analyze fake accounts and highlight signs for further calculation

978-1-6654-0476-1/21/$31.00 ©2021 IEEE

4. Collect dataset for compiling a training and test sample necessary for training and testing machine learning algorithms
5. Choose the optimal machine learning models
6. Build a machine learning model
7. Compare results

IV. Experiments

To achieve the best results, at the first stage of the experiment, the capabilities of the VK API [11] were analyzed and the scope of what was permitted was outlined. The VK API allows you to get only the information that you can get if you just go to the page. If the user profile is closed, then extraordinarily little data can be obtained, so closed accounts were excluded from consideration. Based on personal experience and accounts for which it was known in advance that they were fake, some distinctive features of fake accounts were collected, which were later used to implement machine learning methods. The following signs were considered:

1. User ID;
2. Date of first post;
3. Number of friends;
4. The presence of a face in the profile photo;
5. Number of subscribers;
6. The number of records on the user's page (if more than 1000, then the value was taken as 1000 due to the restrictions imposed by the VK API);
7. The number of groups the user is subscribed to;
8. The number of photos in the profile;
9. Number of audios;
10. Number of videos;
11. Number of subscriptions;
12. Number of gifts;
13. The account has a pseudonym;
14. The number of educational institutions specified in the profile;
15. Link in the first post;
16. Is the city specified in the profile;
17. Is the current place of work or study indicated;

Signs 1-2 made it possible to roughly determine how long ago the account was registered and became active. Since the user ID is unique and is issued by the system to each registered account in order, accounts with a larger ID have a later registration date than accounts with a lower ID. Signs 3-12 indicated the social activity of the account. Signs 12-17 indicated the completeness of filling out the profile data. On these grounds, it was possible to calculate most fake accounts.

The next step was to collect data for training. This data included links to accounts of real users, hereinafter ARU, and links to fake accounts. Also, for additional testing, it was decided to collect data on fake accounts, like the profiles of real people. Such accounts had a completed profile and many photos of one person, but these photos were stolen from the pages of other users or sites. Similar accounts were collected manually. An account was considered fake if the person's profile photo was stolen from another user's real page. The real one of two profiles with the same photo was considered the one in which this photo was uploaded earlier. Thus, 16 links to similar accounts were collected.

The collection of data about real users was done manually. For this, a program was implemented in python [12] to speed up the process. The program displayed the user's page on the screen and when the “YES” button was pressed, added a link to this account to the ARU list and displayed the next user's page. When the “NO” button was pressed, the program simply displayed the next page. The accounts were taken from friends of some non-fake account, usually a friend of the author. The total number of links to the accounts of real people was 2713.

The next step was to collect links to fake accounts. On the VK social network, there are specific groups whose members hide their identity in different ways. For this, fake accounts are used. Accordingly, accounts with the highest creation date and located in such groups are presumably fake, so it was they who were unloaded for manual verification. We considered similar groups with more than 50,000 subscribers. From each such group, using the VK API, 1000 links with the highest identifier were unloaded, these were the accounts registered the latest. After the links were uploaded, the accounts were manually checked for various errors. The signs by which the authenticity of the account was determined were the fullness of the profile, the number of friends and the presence of a photo of the person in the photo in the profile. In this case, closed accounts were also excluded from consideration due to the impossibility of determining their reliability. The total number of links to fake accounts was 3456.

After the formation of the list of links, it was necessary to obtain from these links the signs that had already been selected. To unload features, a program was written that, through the VK API [11], unloaded features from the VK server and wrote them to a csv file. Thus, a dataset of 6169 rows was collected.

Next, it was necessary to select machine learning methods for subsequent classification. The following methods were chosen:

1. Gaussian naive Bayes
2. Bernoulli naive Bayes
3. Support vector machine
4. Decision tree method
5. The random forest method
6. MLP
7. Sequential neural network

Methods 1-4 were implemented using the scikit-learn library [13]. The classifiers from this library were taken in standard variation. Methods 5 and 6 were also implemented using the scikit-learn library, but the classifiers were further tuned.

The configuration of the random forest algorithm was taken from research [5]. The number of trees was 1000.

MLP consisted of 3 layers. The first and third layers had 100 neurons, the middle layer had 500 neurons.

Method 7 was implemented using the Keras library [14]. Classifier 7 was taken in the configuration shown in Fig. 1.

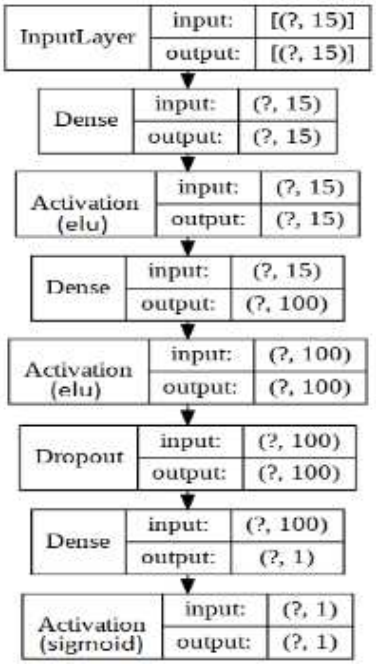

Fig. 1. Sequential neural network model

Features such as user id and date of first post were excluded from consideration since they had a negative impact on the classification accuracy.

All these methods were trained with 5000 rows from the original dataset. Accuracy, recall, and precision were used to assess the effectiveness of the algorithm (Fig. 3). An error matrix was also used (Fig. 2).

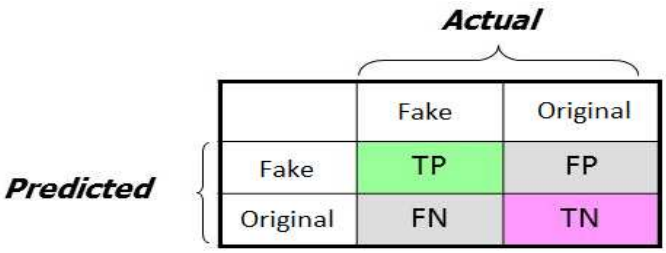

Fig.2. Error matrix

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

Fig. 3. Formulas for calculating recall, accuracy and precision for one class.

V. Research Result

As a result of the experiment, algorithms for calculating fake accounts were trained and tested. Testing took place on the remaining 1,169 rows from the dataset and on pre-assembled fake accounts like the PIU. The overall calculation accuracy is shown in the diagram (Fig. 4). All methods showed an accuracy greater than 80%. The random forest method showed the highest accuracy, its accuracy was 97%. For a more competent assessment, such methods of assessing accuracy as recall and precision were used.

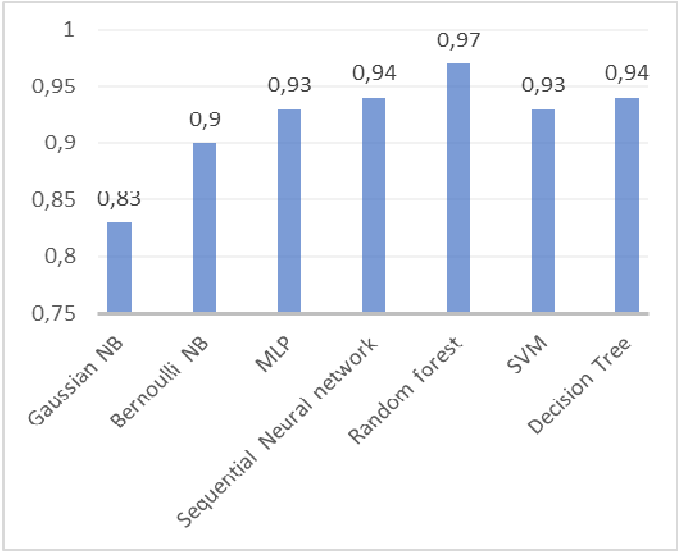

Fig. 4. Overall Accuracy

The results of the classification accuracy of Gaussian naive Bayes were considered separately, as it had the lowest accuracy. As can be seen from the diagram (Fig. 5), Gaussian naive Bayes does a poor job at computing ARU, confusing them with fake ones. Of the total number of ATMs, the classifier designated as genuine only 63%.

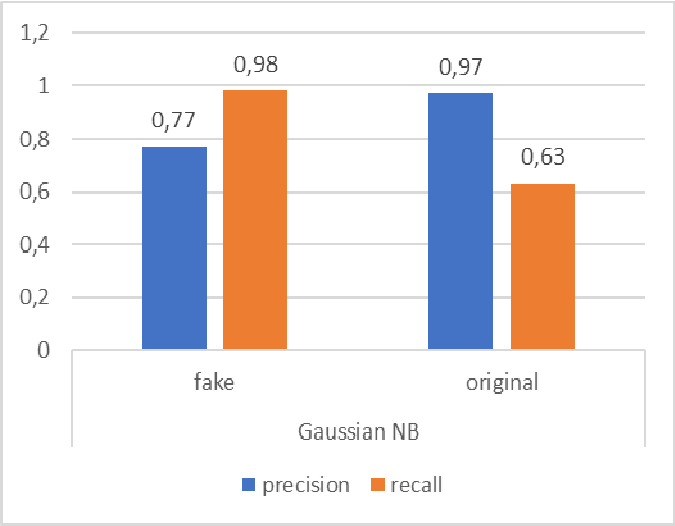

Fig. 5. Gaussian naive Bayes

Bernoulli's naive Bayes results were also considered separately (Fig. 6), since he was the worst in identifying fake accounts.

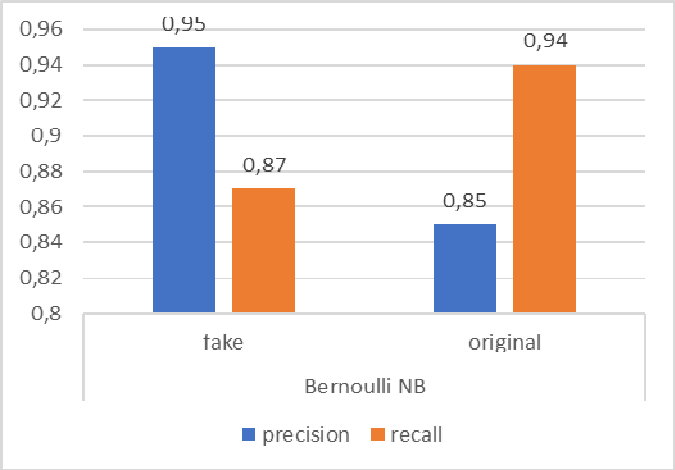

Fig. 6. Bernulli naïve Bayes

Bernoulli naive Bayes has successfully figured out 87% of fake accounts of the total. This method calculated the ARU quite well and correctly determined 94%.

The rest of the methods were considered in more detail (Fig. 7). The random forest algorithm is the best at both detecting fake accounts and detecting ARU. The next most accurate is the decision tree method. The rest of the methods were less accurate both in identifying fake accounts and ARU, but they all showed good results.

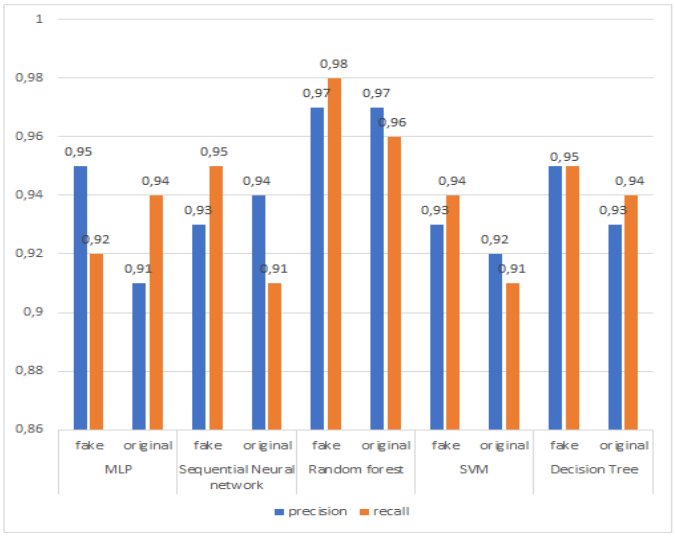

Fig. 7. Taking a closer look at machine learning methods

The methods were also tested on fake accounts like ARU. In fig. 8 shows the percentage of successfully calculated similar.

As you can see from the diagram, Gaussian naive Bayes identified the most fake accounts. Next up are MLP, the random forest method, and the decision tree algorithm. The rest did a little worse.

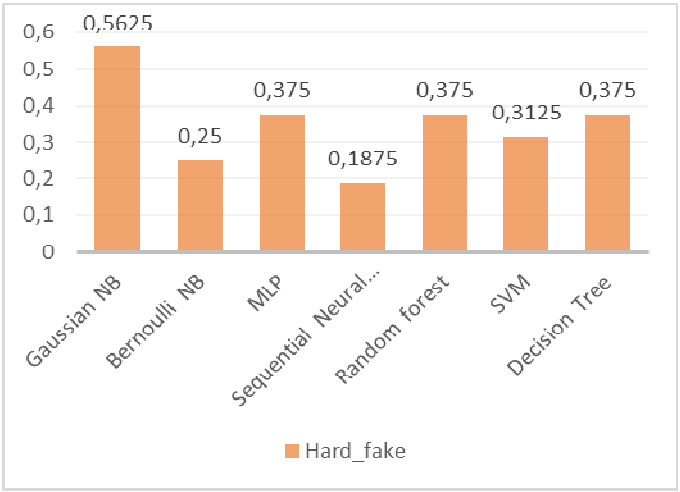

Fig. 8. Accuracy of calculating accounts similar to ARU

VI. Conclusion & Future Work

Based on the results of the study, we can conclude that the random forest algorithm showed the best results for detecting fake accounts, the overall accuracy of this method was 97%. It successfully copes with both the identification of fake accounts and ARU, showing good calculation accuracy. However, this method is not particularly good at detecting fake accounts like ARU. Gaussian naive Bayes is better at this task. But Bayes has a high margin of error and often mistakes real users' accounts as fake, so it may make sense to use this method to detect fake accounts in cases where the accuracy of ATM detection is not important.

Now, all the machine learning methods used in this study are poor at detecting fake accounts like ARU. In the future, it is necessary to expand the list of fake accounts by adding fake accounts like ARU.

Also, due to the VK API limitation on the number of requests per second, working with a large amount of data takes long time intervals.

This research can be useful:

1) Researchers. This research will provide an opportunity to filter fake accounts and not take them into account in various kinds of social media research.
2) Advertisers. Using the results of this study, the advertiser will be able to verify which part of the community is genuine and which is not.
3) Ordinary users. If the user knows that they have a fake account. He will treat him with caution, which may help him avoid fraud.

REFERENCES

[1] D. L. Hoffman and T. P. Novak, "Why Do People Use Social Media? Empirical Findings and a New Theoretical Framework for Social Media Goal Pursuit," *SSRN Electron. J.*, 2012, doi: 10.2139/ssrn.1989586.

[2] L. Khadartseva and V. Kaytmazov, "SOCIAL MEDIA IN BUSINESS," in *Communications*, Hoboken, NJ, USA: John Wiley & Sons, Inc., 2017, pp. 117–129.

[3] D. A. Vasilyevna and P. E. Germanovna, Metodika Vychisleniya Fal'shivyh Akkauntov V Social'nyh Setyah [Methodology For Calculating Fake Accounts In Social Networks], *Informacionnaya Bezopasnost' Regionov* [Information Security of Regions], vol. 3, no. 3 (20), pp. 16–19, 2015.

[4] P. Yana, L. Andrey, and C. Ekaterina, "identification of the person in the internet network," *Sinergiya nauk* [Synergy of Sciences], vol. 22, pp. 954–961, 2018.

[5] A. Kaveyeva and K. Gurin, "'VKontakte' fake accounts and their influence on the users' social network," *Zhurnal Sotsiologii i Sotsialnoy Antropol. (The J. Sociol. Soc. Anthropol.*, vol. 21, no. 2, pp. 214–231, Jun. 2018, doi: 10.31119/jssa.2018.21.2.8.

[6] S. Sheikhi, "An Efficient Method for Detection of Fake Accounts on the Instagram Platform," *Rev. d'Intelligence Artif.*, vol. 34, no. 4, pp. 429–436, Sep. 2020, doi: 10.18280/ria.340407.

[7] B. Ersahin, O. Aktas, D. Kilinc, and C. Akyol, "Twitter fake account detection," in *2017 International Conference on Computer Science and Engineering (UBMK)*, Oct. 2017, pp. 388–392, doi: 10.1109/UBMK.2017.8093420.

[8] M. M. Swe and N. N. Myo, "Blacklist Creation for Detecting Fake Accounts on Twitter," *Int. J. Networked Distrib. Comput.*, vol. 7, no. 1, p. 43, 2018, doi: 10.2991/ijndc.2018.7.1.6.

[9] A. Gupta and R. Kaushal, "Towards detecting fake user accounts in facebook," in *2017 ISEA Asia Security and Privacy (ISEASP)*, Jan. 2017, vol. 1, pp. 1–6, doi: 10.1109/ISEASP.2017.7976996.

[10] P. Pourghomi, M. Dordevic, and F. Safieddine, "Facebook fake profile identification: technical and ethical considerations," *Int. J. Pervasive Comput. Commun.*, vol. 16, no. 1, pp. 101–112, Jan. 2020, doi: 10.1108/IJPCC-06-2019-0049.

[11] "VK API," *21 November 2020*, 2020. https://vk.com/dev/methods.

[12] "Python," *21 November 2020*, 2020. https://www.python.org/.

[13] "scikit-learn," *21 November 2020*, 2020. https://scikit-learn.org/stable/.

[14] "Keras," *21 November 2020*, 2020. https://keras.io/.