FRAUD DETECTION SYSTEM BASED ON ARTIFICIAL IMMUNE SYSTEM

Vitaly Krokhalev

Siberian State University of Telecommunication and Information Science,
Novosibirsk, Novosibirsk Oblast, Russian Federation

ABSTRACT

Nowadays, one of the most important problems for financial companies is fraud related to online transactions. It is becoming increasingly sophisticated and advanced, leading to financial losses on the part of both customers and companies. Based on this, my company was tasked with creating a fraud detection system that is scalable and adaptable to change. This research aims to create a solution that can be used to identify differences in customer behavior patterns and detect fraud. The artificial immune system model proposed in this article, combined with certain informative features, is simple to implement and can describe customer behavior patterns.

KEYWORDS

Fraud Detection, Artificial Immune System, Informative Features, Machine Learning, Information Security.

1. INTRODUCTION

A fraud transaction detection system, a fraud monitoring or antifraud system, is a system designed to evaluate financial transactions on the Internet for suspicion of fraud and offer recommendations for their further processing. Currently, a significant number of monitoring systems are aimed at detecting certain fraud signatures. This approach detects only fraudulent operations described in the signatures, and for a fairly short period of time - attackers adapt, find new vulnerabilities and use new tools for the next attacks.

The most pressing and significant problem in building a fraud detection system is the imbalance of classes. The array of analyzed data is very large, assorted, and imbalanced, so there is a problem of processing a very large volume of data. According to statistics, the number of fraudulent transactions per total number of transactions does not exceed 0.01%. Such a huge imbalance significantly complicates the detection of fraudulent operations. It follows that the problem of detecting such operations (the problem of classification) should be solved in the context of anomaly detection.

In anomaly detection, we assume that there is a "normal" distribution of data points, and anything that sufficiently deviates from this distribution is an anomaly. If we transform the classification problem into an anomaly detection problem, we can consider the majority class as a "normal" distribution of points, and the minority as anomalies.

The anomaly detection problem is a complex problem and belongs to the class of poorly formalized. Various heuristic algorithms, including bioinspired algorithms, are used to solve such

David C. Wyld et al. (Eds): DMML, SEAS, ADCO, NLPI, SP, BDBS, CMCA, CSITEC - 2022
pp. 83-94, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.120707

problems. Heuristic algorithms do not guarantee finding the optimal solution, but they allow one to obtain solutions of acceptable quality quite fast. Bioinspired algorithms are based on the use and modeling of the principles of organization and functioning of various natural systems, such as neural networks of the brain, the immune system, the evolution of living organisms, the genetic laws of heredity and variability, and swarm intelligence.

Artificial Neural Network (ANN) models are some of the most common bioinspired algorithms nowadays. ANNs allow to solve very complex data processing problems, so their application in the context of anomaly detection was considered first. In "Credit Fraud. Dealing with Imbalanced Datasets"[1], the author implements a neural network with three fully connected layers, describes in detail various errors connected with the class imbalance, and shows the necessity of data preprocessing before feeding into a neural network. This means that using the ANN model is not the most preferable solution for our problem.

Alternatively, we can consider a relatively new class of bioinspired algorithms, the Artificial Immune Systems (AIS) class. AIS continue to be actively studied and are increasingly used in various fields, including information security. Some AIS models, such as the negative selection model, show high efficiency in finding anomalies.

One of the main differences between AIS models and ANN models is their learning. ANN learning is implemented by a special algorithm focused on the type of task and the type of a given ANN, by presenting images of different classes followed by adjusting the weights of the links. In AIS learning is implemented by creating recognition elements - detectors, by positive or negative selection of information units of images of only one class "of their own". Thus, the AIS model can work well in conditions of strong class imbalance.

In the general case, the negative selection model is limited by the possibility of binary classification "friend or foe". However, the task of monitoring fraud involves the use of a degree of confidence in the prediction, which characterizes the probability that a given transaction is fraudulent.

While using the AIS model to create a fraud detection system, the following questions arose:

- How to implement a confidence analysis of the prediction made by the AIS;
- How to identify informative features based on customer transaction information.

This article considers the selection and creation of informative features for the AIS model within the frame of solving the task of fraud transactions detection.

2. LITERATURE SURVEY

The literature review showed that a significant number of publications have been devoted to AIS, noting the successful application of AIS in various application areas. Although research is moving forward, detailed information on the application of AIS in fraud detection is currently insufficient, as financial companies do not disclose specific details of the implementation of antifraud systems.

The article "Artificial immune systems: review and current state"[2] reviews the current state of artificial immune systems. Their problems, advantages and disadvantages, current developments in the field of artificial immune systems, and their areas of application are considered. This article also provides a comparative table of the following heuristic bioinspired algorithms:

- Genetic algorithms
- Neural networks
- Artificial immune systems

For each of the algorithms, the corresponding components are listed. The table shows that the main families of algorithms from the class of biological algorithms that are used have a lot in common. The article notes a significant advantage of AIS over genetic algorithms and artificial neural networks is the ability to learn and the availability of memory.

The article "How to choose the antifraud system?"[3] considers the main principles of modern antifraud systems, actual problems of antifraud systems, the efficiency of different methods of fraud detection, as well as the most popular and effective machine learning algorithms. The following problems are cited as the most significant problems:

- The imbalance of legitimate and fraudulent transactions; •The necessity of detecting fraudulent activities in real-time.
- Dynamically changing fraudulent behavior;
- Significant differences in customer behavior patterns.

The article notes the particular effectiveness of machine learning methods compared to signature rule-based approaches, noting the widespread use of machine learning-based antifraud systems in recent years. A comparative table of key machine learning algorithms is provided. Algorithms were compared based on their frequency of use and the following evaluation criteria:

- The algorithm should have high accuracy in detecting fraudulent actions when processing large amounts of data - "accuracy";
- The algorithm must cover the maximum number of possible fraud scenarios - "coverage";
- The algorithm must be the least expensive in terms of both time and money - "cost".

The article "Machine learning against fraud in banking"[4] describes the historical approach and practical experience in detecting fraudulent transactions, describes the basic principles that must be followed when solving the problem of creating a fraud detection model. In addition, the main problems encountered in solving this problem are considered. According to the cybersecurity service practice described in the article, in order to build an effective fraud detection system, it is necessary to create additional features describing customer behavior in addition to existing transaction data. It is noted that a wide range of different aggregations and mathematical functions are usually used when creating features: percentiles, averages and deviations, sliding windows, and many others. Part of the article is devoted to metrics for estimating model efficiency.

The article "Credit card fraud detection using neural network and geolocation"[5] proposes a system based on a neural network that facilitates the detection of fraudulent transactions by analyzing the location of the customer, while taking into account the structure of his expenses. The main value in the article is the classification and description of various fraudulent schemes.

The article "A Survey of Credit Card Fraud Detection Techniques: Data and Technique Oriented Perspective"[6] describes current technologies in credit card fraud detection, lists and compares a large number of different machine learning techniques (including most AIS algorithms), and classifies these techniques into two main approaches to fraud detection: abuse detection (using teacher learning) and anomaly detection (using teacherless learning).

3. PROPOSED DESIGN

3.1. Implement a Confidence Analysis

The analysis of the prediction confidence of the AIS is associated with the use of numerical features, which leads to the expediency of using the appropriate detectors - numerical detectors. In this case, the numerical features of the analyzed data are represented by points in the feature space, which can be interpreted as vectors emanating from the origin of the feature space. Numerical detectors are characterized by the coordinates of their centers in feature space and radii. The comparison of the features of the analyzed data with the detectors is carried out on the basis of proximity measures between the corresponding vectors in the feature space.

The information units that the AIS algorithm characterizes as representatives of the "foreign" class must fall into the area of one of the detectors, and the representatives of the "own" class must not fall into any detector. Then the probability that the transaction is fraudulent should be greater than 0.5 if the point is inside the detector. If the point is at the detector boundary, then the probability will be equal to 0.5. We need a function that will match the received number with another number from 0 to 1 (this is what we will interpret as a probability). As such a function, you can use the sigmoid function. When working with detectors, the algorithm will operate on the distances between the test point (the current transaction) and the center of the detectors. Let's define the extreme cases: if the point hits the center of the detector directly, then the result is 0, that is, the sigmoid argument is -∞. When hitting the boundary of the detector, the argument is 0, and when moving away from the detectors, the argument is +∞. Thus, the closer the point is to the center of the detector, the lower the degree of transaction reliability.
The general formula will be as follows:

$$p = sigmoid\left((1-y)*\frac{1}{k}\sum_{i=1}^{k} abs(R_i - r_i)*r_i + y*\frac{-(R-r)}{r}\right)$$

where $y = \{0, 1\}$ is a binary identifier of a point belonging to some detector.

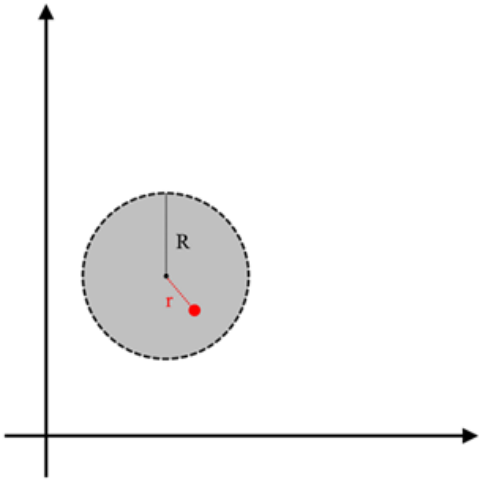

Figure 1. Pont entering the detector area

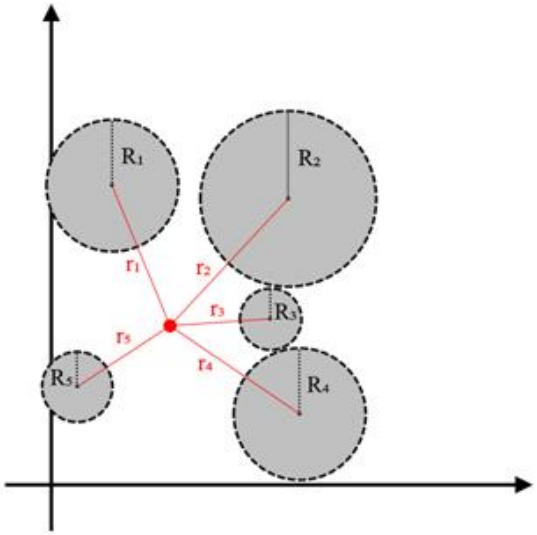

Figure 2. The point is outside the detector areas

A common example of a sigmoid function is the logistic function defined by the formula:

$$S(x) = \frac{1}{1 + e^{-x}}$$

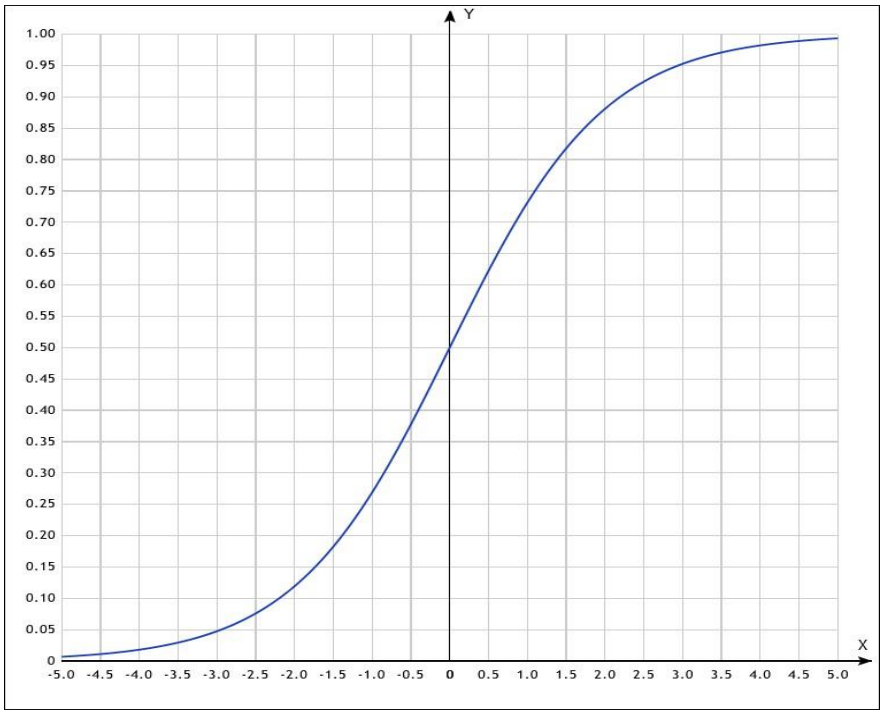

Figure 3. Sigmoid curve of the logistic function

When the behavior of the function changes, the end result will also change: when the point approaches the center of the detector or moves away from the detector boundary, the decrease or increase in the degree of prediction confidence will change more sharply. For the logistic function, you can enter the parameter α, which affects the nature of the function, making it flatter

or sharper. This approach makes it possible to perform a more precise adjustment of the detector parameters, which is optimal for a specific task.

The formula will be as follows:

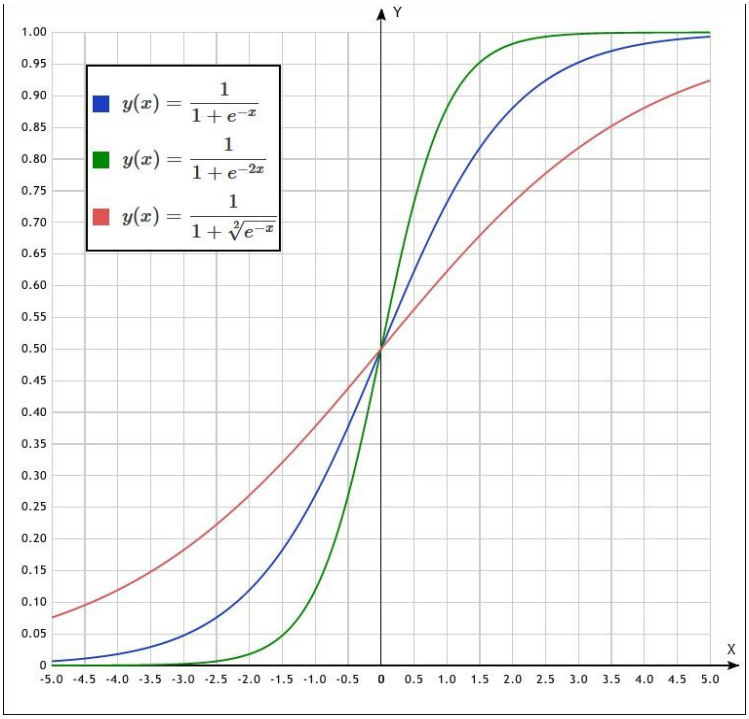

Figure 4. Sigmoid curve of the logistic function with different α

3.2. Creating Informative Features

The application of AIS in the field of fraud detection is described quite often, but the bulk of the information falls on the comparison of AIS with other algorithms, excluding the details of system implementation, in particular, the selection and creation of informative features. We will consider the features for the AIS model in more detail.

Denote the information about the transaction that comes into the system. The data structure representing information about a transaction includes the following fields:

- id (string) - unique identifier of the transaction
- merchant_id (string) - unique merchant identifier
- service (string) - unique service name (mobile bank, PayPal, etc.)
- amount (floating point number) - transaction amount
- currency (string) - currency code
- createdAt (date and time) - date and time of transaction initiation
- customer_id (string) - unique identifier of the client
- status (string) - unique identifier of the current transaction status
- history (history object array) - contains the history of the transaction statuses

History object includes the following fields:

- status (string) - unique identifier of the current transaction status
- date (date and time) - date and time of the status assignment

Of the presented data fields, the following ones are informative features:

- merchant_id
- service
- amount
- currency
- time

Some of the already extracted features are of the string type, and some are of the numeric type. The AIS algorithms can handle both the first type and the second, but for simplicity, it would be reasonable to convert the initial and further obtained features into a numeric format. Such a conversion is justified by the fact that it is not important for us, for example, what specific currency or service was used, what matters is the presence of anomalous behavior: that is, the same currency (or service) is used for this particular transaction as before, or it is the first time this happens.

The practice described in the articles shows that in addition to the existing transaction data, it is necessary to create additional features describing the client's behavior. Many new features can be generated from the existing source features by applying various techniques.

To begin with, we have to condition ourselves on the existence of such a concept as time periods. There will be several of them, they are usually needed for the sliding window. Suppose there will be 4:

t_{p1} = 1 hour, t_{p2} = 24 hours, t_{p3} = 7 days, t_{p4} = 30 days, t_p – unspecified period.

Based on the information from the articles, we propose some new features computed on the basis of the original ones.

Amount

- The average value of *amount* for t_{p1} during t_{p2} (t_{p3}), divided by the total value of *amount* for a year (or other long period);
- The average value of *amount* for t_{p2} during t_{p3} (t_{p4}), divided by the total value of *amount* for a year (or other long period).

Service

Percentage of transactions with this *service* for period t_p.

Amount + Service

- Percentage of *amount*, spent using this *service* for the period t_p;
- Average value of *amount*, spent using this service in period t_p, divided by the annual total value of *amount* (or other period).

Currency

Percentage of transactions using this particular *currency* for the selected period.

Currency + Amount

For each currency used you can add the features from the *Amount* section.

Tnx (transactions)

The average number of transactions in period t_{p1} (t_{p2}) during t_{p2} (t_{p3}). This value can be normalized if the variation of transaction frequency is too different from client to client.

Tnx + Amount

The average number of the amount per transaction during the period t_p, divided by the total amount per year (or other long period).

Time

For time, the following feature is suitable: whether the time of transaction initiation falls into the confidence interval. To do this, you need to do a little preprocessing of the data: calculate the average, deviation, and set the probability for the interval. The peculiarity is that in this case the mean and deviation are calculated in a different way because for the time it is necessary to calculate the periodic average by the following formula:

$$\mu_{vM}(D) = 2\tan^{-1}\left(\frac{\sum_{t_j \epsilon D}\sin(t_j)}{\left(\sqrt{\left(\sum_{t_j \epsilon D}\cos(t_j)\right)^2 + \left(\sum_{t_j \epsilon D}\sin(t_j)\right)^2} + \sum_{t_j \epsilon D}\cos(t_j)\right)}\right)$$

Then the deviation is calculated:

$$\sigma_{vM}(D) = \sqrt{\ln\left(\frac{1}{\left(\frac{1}{N}\sum_{t_j \epsilon D}\sin(t_j)\right)^2 + \left(\frac{1}{N}\sum_{t_j \epsilon D}\cos(t_j)\right)^2}\right)}$$

The calculated values are substituted to calculate the von Mises distribution as follows:

$$x_i^{time} \sim vonmises\left(\mu_{vM}(S_{per}), \frac{1}{\sigma_{vM}(S_{per})}\right)$$

The final formula for calculating the von Mises distribution will be as follows:

$$f(x \mid \mu, k) = \frac{e^{k\cos(x-\mu)}}{2\pi I_0(k)}$$

Then, by setting the probability, we find out whether the time of initiation of the given transaction is within the confidence interval. To perform mathematical operations, the time is converted to floating point numbers.

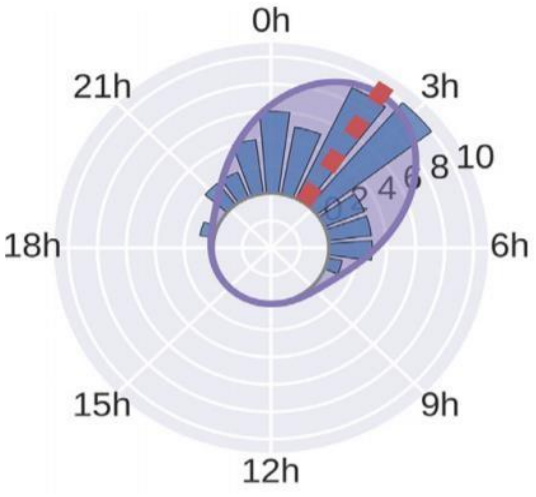

Figure 5. A visual representation of the described case. The red line is the periodic average, the purple outline is the resulting distribution. The columns represent transactions.

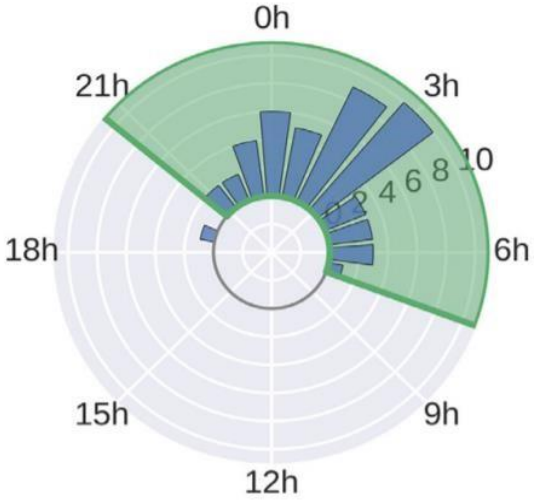

Figure 6. A visual representation of the described case. The transaction columns inside the green sector fall within the confidence interval

4. DISCUSSION

For the above features, it is assumed that the number of transactions in the history of each client is sufficient. However, in cases where the user has a small number of transactions, false triggers of the system become possible. For some features, this problem is solved by rationing, for example, in the case of the amount. However, for some features, there are separate cases, for example, for such a feature as the percentage of transactions with use (which is already a normalized value). If the user's history contains four transactions using one service, and a new transaction will use a different service, in this case, the percentage of use for the new service will be zero, which can affect the model prediction, although obviously there is a short history of

transactions. In this case, an additional summand can be introduced for such values. An example is shown in the following table.

Table 1. Example with an additional summand

a	0,01				
k	0,05				
tnx	1	2	3	4	5
service	s_1	s_1	s_1	s_1	s_2
% s_1	100	100	100	100	80
s_1 count	1	2	3	4	4
% s_2	0	0	0	0	20
s_2 count	0	0	0	0	1
% s_1'	99,9500	99,9750	99,9833	99,9875	79,9900
% s_2'	4,9500	4,9750	4,9833	4,9875	22,9900
softmax s_1	0,7211	0,7211	0,7211	0,7211	0,6388
softmax s_2	0,2789	0,2789	0,2789	0,2789	0,3612

Table 2. Description of the used variables

tnx	Number and quantity of transactions
service	Type of service
s_1 count	Number of transactions with the first service type
s_2 count	Number of transactions with the second service type
% s_1, % s_2	Initial current percentage of transactions with this service
% s_1, % s_2	Recalculated current percentage of operations with this service
softmax	Normalized recalculated current percentage of operations with this service

The formula for recalculation from $s_i\%$to $s_i\%'$:

$$s_i^{\%\prime} = s_i^{\%\prime} - \frac{k * s_{top} - s_{curr} + a}{tnx}$$

Where s_{top} – is the number of operations with the most frequent view, s_{curr} – is the number of operations with the current view, tnx – total number of transactions, k, a – constants less than one. Strictly speaking, after recalculating the percentage, it is no longer percent, since its sum is not equal to 1, so it should be further normalized in some way. In this case, this function $softmax$ was given as an example.

5. FUTURE ENHANCEMENT

In order for the AIS to be more accurate, it is possible to select the optimal parameters for calculating the degree of confidence. For example, by sorting through various options for the sigmoid parameter, you can find the optimal value at which the degree of prediction error will be minimal.

When training AIS, you can change the size of the window that analyzes the data, which affects the distribution and size of the detectors. It is also possible to reduce the degree of prediction error by examining different window sizes. For a more detailed analysis of the transaction, you can enter learning for different time periods, such as a regular day for a certain time, a regular day of the week, a day of the month, and so on. The AIS can be retrained over time so that more recent data is used and outdated data no longer affect the prediction result.

In addition to the listed features, we can add another group of features with *merchant_id* by analogy with the previous features. For example, *Tnx + Amount*, taking into account transactions with a specific merchant.

6. CONCLUSION

Fine-tuning of detectors and the proposed methods of obtaining informative features necessary to build a fraud detection system allow to describe the client's behavior in detail. The use of additional features helps to identify complex or previously unknown fraud patterns using all available parameters, as well as to adapt to changing fraud schemes.

ACKNOWLEDGEMENTS

I express my gratitude to my supervisor (Ivan Nechta), who helped to correct various shortcomings and gave a review on this article. Special gratitude is expressed to the organizers of ADCO for giving the opportunity to publish this work.

REFERENCES

[1] Kaggle, "Credit Fraud. Dealing with Imbalanced Datasets". [Online]. Available: https://www.kaggle.com/janiobachmann/credit-fraud-dealing-with-imbalanced-datasets.

[2] Chernyshev Yu.O., Grigoriev G.V., Ventsov N.N., "Artificial immune systems: review and current state," in *Software products and systems*, vol. 4, pp. 136-142, 2014.

[3] Nikita Andreyanov, "How to choose the antifraud system?" in *Journal IT-Manager*, 2019. [Online]. Available: https://www.it-world.ru/cionews/manage_secure/148780.html

[4] Andrey Pinchuk, "Machine learning against fraud in banking," in *Journal IT-Manager*, 2017. [Online]. Available: https://www.it-world.ru/cionews/manage_secure/118786.html

[5] Aman Gulati, Prakash Dubey, MdFuzailC, Jasmine Norman, Mangayarkarasi R, "Credit card fraud detection using neural network and geolocation," in *IOP Conference Series: Materials Science and Engineering*, pp. 1-6, 2017.

[6] SamanehSorournejad, Zahra Zojaji, Reza Ebrahimi Atani, Amir Hassan Monadjemi, "A Survey of Credit Card Fraud Detection Techniques: Data and Technique Oriented Perspective," *Cornell University*, 2016. [Online]. Available: https://arxiv.org/ftp/arxiv/papers/1611/1611.06439.pdf

AUTHOR

Vitaly Krokhalev M.S. at the Siberian State University of Telecommunication and Information Science in Computer Science; Android developer at Elementpay international company. His area of interest includes software development for Windows, Linux, and Android.