Computers & Security 120 (2022) 102786

Contents lists available at ScienceDirect

Computers & Security

journal homepage: www.elsevier.com/locate/cose

Reducing false positives in bank anti-fraud systems based on rule induction in distributed tree-based models

Ivan Vorobyev[a,*], Anna Krivitskaya[b]

[a] HSE University, Russian Federation
[b] Lomonosov Moscow State University, Russian Federation

ARTICLE INFO

Article history:
Received 27 September 2021
Revised 14 March 2022
Accepted 2 June 2022
Available online 6 June 2022

Keywords:
Fraud detection
Payment card fraud
False positives
Rule Induction
Feature engineering

ABSTRACT

Fraud detection in bank payments transactions suffers from a high number of false positives. To deal with this problem, we introduce a rules generation framework for a fraud-detection system – an automatic rules generation using distributed tree-based ML (machine learning) algorithms such as Decision Tree, Random Forest and Gradient Boosting, where the components of expert rules are used as the features for the model. This approach is a combination of statistical and expert-based approaches. We apply it to the bank's card transaction data. Our dataset covers February 2021 and consists of more than 20 mil. records including information on clients, transactions, and merchants. The autogenerated rules were aimed at improving FPR (false positive rate) business-metric. The framework was tested in a real fraud-monitoring system of large bank throughout half of the year. The rules obtained using this framework proved to be satisfactory efficient while having tangible business effect.

1. Introduction

1.1. Motivation to the study

As fintech and e-commerce thrive, the more and more bank payments and money transfers are facilitated through the online channels which are faster, more convenient, and safer for the health in a coronavirus era. According to Mastercard global consumer study 2020,[1] 8 out of every 10 Mastercard users all over the world make use of contactless payment methods.

The data on conducted transactions is accumulated in databases of banks, e-commerce platforms, and other e-commerce industry players. For the year 2017 a Visa payment processing capability was as high as 75 000 transactions per second (Zeng, 2018). The proper use of that data allows corporations to improve their operational performance and customer experience. However, that much data is impossible to handle manually. Hence, the corporations are forced to build Big Data infrastructure and utilize Machine Learning (ML) algorithms.

Developing mechanisms to protect client funds against fraudsters is a part of banking and e-commerce corporations' strategy for improving customer experience. As payments have moved to online channels, the fraudsters have adapted as well, bringing issues of protecting the funds circulating in online channels to the fore. According to SmartMetric, in 2018, the global losses from payment fraud were higher than $24bn.[2]

Despite the difficulties while detecting fraud (driven by the fraudsters imitating normal clients' behavior and using social engineering techniques), the fraud-detection systems of large corporations allow to detect and prevent up to 98% of fraud cases (Carminati et al., 2015). In terms of ML metrics, this means attaining a high recall.

Nowadays, an issue of a large number of false positives (FP) comes to the fore (Zoldi, 2015), stating the problem of low precision, or high false positive rate (FPR). On average, there is only 1 fraudulent transaction out of 5 transactions blocked, and every 6th user was blocked mistakenly over the year (Wedge et al., 2019).

False positives lead to money losses of corporations on investigating cases and contacting the client, as well as to an increased pressure on a call-center, and even to revenue losses due to declined transactions. Considering one of the possible response strategies – giving a call to the client – the cost of managing one transaction blocked ranges from 1.5 euros (according to the bank that provided us with data for our research) to 5 euros (European Central Bank, (Baesens et al., 2021b)). As for the e-commerce industry players, according to Merchant Risk Council's 2017 Global

* Corresponding author.
E-mail address: vorobyev-ivan@yandex.ru (I. Vorobyev).

[1] https://www.mastercard.com/news/press/press-releases/2020/april/mastercard-study-shows-consumers-globally-make-the-move-to-contactless-payments-for-everyday-purchases-seeking-touch-free-payment-experiences/

[2] https://www.businesswire.com/news/home/20191223005414/en/

https://doi.org/10.1016/j.cose.2022.102786
0167-4048/

Fraud Survey,[3] one online retailer declines on average 2.6% of orders, at least 10% of which should have been accepted. According to Riskified,[4] false positives in that case lead to a loss of 6% of the revenues.

In addition, the more demanding the clients become to the services quality, the more false positives damage the company's reputation and decrease the customers' loyalty. According to Javelin Strategy,[5] 61% of users who didn't manage to conduct a transaction due to the false positive blocking, cut down on card usage or stopped using them all together.

1.2. The purpose of the paper

This paper aims at searching for the approach to improve the efficiency measures of bank fraud detection process, which consist of two key metrics: 1) fraud basis point (FBP), accounting for fraudulent transactions not blocked by fraud-detection system, and 2) false positive rate (FPR), accounting for the fraction of false positives among all the alarms generated by the system.

The work is concentrated on FPR metric. We propose a method to reduce FPR while having no deterioration on FBP metric.

1.3. Data and methods

We search for the solution that combines expert and statistical approaches to fraud detection in such a way that the resulting algorithm incorporates the advantages of both and performs well as measured by the key business metrics while being highly interpretable. At the same time, the solution should be easy to implement and integrate into existing fraud-detection systems used by the corporations.

We utilize the rules induction techniques. We generate new rules by implementing tree-based ML algorithms, the features of which consist of the components of existing experts' rules. The resulting rules are aimed at identifying cases that currently are incorrectly classified as fraud.

The data for our research includes characteristics of transactions and clients, as well as the set of expert rules that form the decision-making logic.

1.4. Definition of fraud

By the concept of fraud we mean a case of money theft from a client by professional fraudsters. We classify fraud into two main types depending on techniques used by fraudsters – social engineering, when fraudsters persuade a client to take actions that enable them to steal the money, and others, not involving a client's participation.

Fraud, according to (Baesens et al., 2015), is characterized by the following specific characteristics:

- Fraud is uncommon, compared to the legitimate transactions
- Fraud is well-considered and organized
- Fraudsters try to conceal their actions, pretending to be the normal users
- Fraud patterns are dynamic, changing over time
- Fraudsters can work as a group

1.5. The novelty of the proposed approach and its practical applications

From the scientific perspective, our research contributes to the literature on fraud detection methods. We propose an approach that combines expert-based and ML approaches such that features to the model are derived from the expert algorithms and the output of the model imitates the algorithms which experts could have constructed themselves. The application of statistical methods for fine-tuning experts' algorithms improves the efficiency of fraud-detection measures, primarily in terms of number of false positives.

Also, we discuss the practical applications of our approach applied to fraud-detection system. According to (Pant and Srivastava, 2021), there is a problem with academic researchers as they lack business context understanding on how real fraud-detection systems work and what are the costs of blocking legitimate or not blocking fraudulent transactions. Thus, our discussion contributes to that type of knowledge in academia.

Regarding practical applications, the outcome of the proposed algorithm is represented in the form of decision rules that can be easily integrated into fraud-detection systems the corporations use nowadays. The method is relevant for the organizations that utilize such systems to prevent fraud (e.g. banks, payment systems, e-commerce platforms, insurance companies, fiscal authorities), as well as for other companies fostering automatization of decision-making processes on the basis of decision rules derived from Big Data. In (Schneider and Xhafa, 2022), more details for the eHealth application area are provided.

Apart from the description of an algorithm, we also provide some detail on how to evaluate it in real time and match the actual real-time results with those obtained on the training set. Finally, we suggest a technique to evaluate the net business effect of it, given the fact that there is already some level of precision and recall achieved in corporations with the use of their current decisions.

1.6. Results

As a result of implementing the proposed framework in a bank, we tested rules targeted at detecting false positives or false negatives in a real industrial antifraud system. The rules performance was evaluated in terms of their classification quality and net financial effect. The best-performing rules were then added to the antifraud system.

Although we indicated a need for further improvements, we have already achieved fairly good performance metrics. The average precision (measured as true positives to sum of true positives and false positives) of the rule is 50% on test sample though 10% in online mode. The average recall of a rule on test sample is 0.6%.

This paper is further organized as follows: Section 2 is a literature review, Section 3 provides detail on the data we used and the field we work in, Section 4 describes the algorithm we implement, Section 5 presents the results of modeling and Section 6 concludes.

2. Literature review

Fraud detection and prevention is of interest to both business and academia. While in 2015, 16,000 scientific papers were published on the topic, in 2021 it was 1.5 times more.[6]

Fraud detection algorithms could be classified into expert and statistical approaches. In the first case, fraud is detected on the basis of rules created manually by experts who analyze typical fraud

[3] 2017 MRC Global Fraud Survey. https://www.merchantriskcouncil.org/resource-center/surveys/2017/2017-mrc-global-fraud-survey?authResult=success

[4] Shalhevet Zohar. The True Cost of Declined Orders. 2018. https://www.riskified.com/blog/true-cost-declined-orders/

[5] Pascual A. Future proofing card authorization. 2015. https://www.javelinstrategy.com/coverage-area/future-proofing-card-authorization

[6] Source: количество исследований в Google Scholar по ключевым словам 'fraud detection'

patterns. In the second case, the Artificial Intelligence (AI) methods, especially Machine Learning (ML), are applied to reveal fraudulent operations.

As stated above, the high false positives rate remains one of the key problems in the field. The recent research that seeks to resolve the problem is concentrated mainly on statistical approaches. These include the feature engineering techniques that increase the efficiency of models in terms of FPR such as automated feature engineering (Wedge et al., 2019); the deep neural networks that help to automate the decision-making process and prove to provide sufficient classification quality in fraud-detection (Carrasco and Sicilia-Urbán, 2020)), the classical ML algorithms including clustering (Liang et al., 2015) and classification models (Severino and Peng, 2021), etc.

A very few works concern the problem of how to efficiently combine expert knowledge and AI. Most of the articles that try to account for both approaches are concentrated on the development of AI analytical and data visualization tools to assist fraud experts (Sun et al., 2020; Leite et al., 2020) and explainable AI which fraud experts are believed to trust more than the black-box models (Cirqueira et al., 2021), as well as on the modeling and data engineering techniques which should complement expert-driven traditional approach (Baesens et al., 2021). There are also papers based on the idea of expertise-based feature engineering as a means to extend the typically generated recency, frequency, and temporal features (Hsin et al., 2021; Xie et al., 2019). One more way to apply experts' knowledge to improve the efficiency of statistical models is proposed in (Rao et al., 2021), where experts produce the set of rules that filter out the noisy unlabeled data from the training dataset.

Our experience, including constructing new features, performing feature selection automatically, clustering merchants' profiles, detecting abnormal behavior of clients and merchants, creating risk scores based on neural networks and so on, proves that the usage of statistical algorithms does help fraud experts to reduce false positives. But we believe that the performance of fraud-detection system as a whole, accounting for both experts and statistical approaches, can be further improved combining two types of intelligence – natural and artificial – in a more automated way.

In this work, we propose to complement the experts' knowledge with AI through application of the rule induction techniques. So far the rules induction was seen to be a data mining technique that helps to reveal hidden patterns in data. The resulting association rules were those used as a supportive tool for experts' decision making. For example, (Xie et al., 2019) imply rules induction to engineer new features over the set of rules and further use those in a Random Forest classifier. However, all the rules here were generated manually based on expertise accumulated by analytics, not automatically. (Sadgali et al., 2021) proposed ML rules generation approach to assess the risk level associated with each transaction. They induce fuzzy association rules sets based on Apriori algorithm and then score transactions depending on the share of rules in the set the transaction is consistent with.

Our approach differs from the association rule mining described above on the basis of how we understand the concept of rule. We aim to derive the ready-to-use expert-like if-then rules in a conjunctive normal form suitable for using in a fraud detection system as they are, with no need for further experts' efforts to interpret and adjust them. The approach that is most closely to ours is of (Youssef et al., 2021). The authors utilize the deep-learning framework CRED (continuous/discrete rule extractor via decision tree induction) to induce the if-else rules in e-commerce fraud detection. The main purpose of applying rule induction techniques was to shed light on the process of forming predictions in black-box deep learning models. The other related work is (Hasanpour et al., 2019) where classification and rule mining were integrated into a rule-based classifier by merging Apriori associative rule induction algorithm with binary Harmony Search rule selection and "Classification Based on Associations" algorithm for building classifier in the form of an If-Then-Else rule list.

For the purpose of our research, we have chosen the rule-based models for rule induction such as Decision Tree, Random Forest and Gradient Boosting. According to (Hasanpour et al., 2019), these models, although not demonstrating the better classification metrics than the customized and more complex ones, are satisfactory in terms of the metrics, computing resources and time spent on training and tuning the model. The last two aspects are particularly relevant for us having billions of data records and applying Big Data technologies stack and distributed ML models.

One more thing that differs our approach is being driven by industrial needs of our company and thus being fully concentrated on the problem of reducing the number of false positives. We select the rules that predict the legitimate transactions, and we propose a custom set of ML and business metrics that correspond to a given task. We explain in detail how our approach can be integrated into the fraud-monitoring system with some given level of precision and recall. Finally, we test the efficiency and scalability of our approach on real data.

While conducting our research, we faced the problems typical for the industry: the large imbalance of classes (according to [5], fraudulent transactions make up less than 0.5% of the sample) and unequal costs of classification errors. The latter one concerns the unequal costs of misclassifying fraudulent and legitimate observations. Especially, the cost of false positives and true positives is fixed at the level of administrative costs for investigating the case and contacting the cardholder, while the cost of false negatives depends on the sum of money stolen by fraudster [13].

The typically used techniques to account for classes imbalance and unequal costs problems are changing the model inputs by either oversampling (Kumari et al., 2019; Baesens et al., 2021a, 2021b), undersampling (Trisanto et al., 2020) or combining both of them; adjusting weights of observations; changing the model outputs by correcting the thresholds (Sheng, 2006); or changing the classification algorithm itself by modifying the existing ones or developing the new ones (Höppner et al., 2021). If the ratio of two types of error costs does not remain fixed within the class, the weighting and changing algorithm approaches work only.

In case of tree-based algorithms, there are several options of modifying cost-insensitive algorithms to the cost-sensitive one: 1) splitting in a cost-sensitive manner, 2) pruning the tree in a cost-sensitive manner, or 3) using an additional cost adjustment function inside the impurity criteria (Sahin et al., 2013).

In our work, we apply undersampling and prune rules in a cost-sensitive way. We have also experimented with classes weights.

3. Application area and data description

3.1. Business metrics of transactional antifraud quality

The business metrics to evaluate the efficiency of fraud detection process in transactional antifraud are influenced by ML and defined by types of transactions in the fraud-monitoring system.

All the banking operations (turnover) that pass through the antifraud system can be divided into 5 categories depending on the system verdict (Fig. 1):

- Fraud_identified – the operations that were blocked by the system and were given feedback by the client that they were actually fraudulent
- Genuine (false positives) – the operations that were blocked by mistake

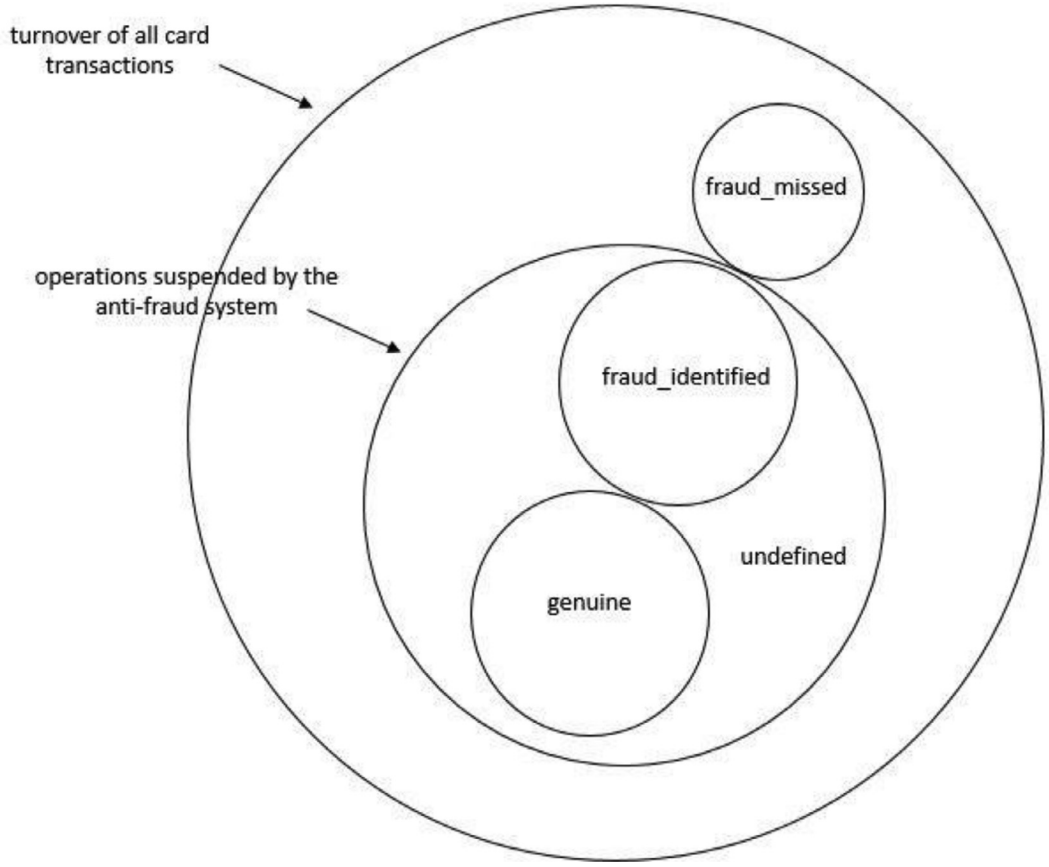


Fig. 1. Types of operations in anti-fraud system.

- Undefined – the operations that were blocked but no feedback from a client was received
- Fraud_missed – the operations that were not blocked, but clients reported they were fraudulent
- Others – the most numerous operations that were approved by the system, and the clients did not report fraud on them («turnover for all card transactions» minus «operations suspended by the anti-fraud system» and minus «fraud_missed» in Fig. 1).

The objectives of the transactional antifraud can be expressed in controlling two business metrics:

$$\text{FBP} = \frac{\text{Fraud\_missed}}{\text{Turnover}} \tag{1}$$

$$\text{FPR} = \frac{\text{Genuine}}{\text{Genuine} + \text{Fraud\_identified} + \text{Undefined}} \tag{2}$$

The main task of the fraud monitoring system is to minimize FBP (fraud basis point) – a metric that corresponds to the amount of fraud missed.

On the other hand, to ensure a positive user experience from consumption of banking products, the anti-fraud team should not block the operations mistakenly. Hence it is important to minimize FPR (false positive rate) – a metric that corresponds to the number of false alarms. This indicator is calculated as a ratio of false positives number to the overall number of triggers, thereby being equal to $(1 - \textit{precision})$ in terms of ML classification metrics.

Additionally, fraud-monitoring experts should try to reduce the number of undefined operations, since it is a blind spot where experts cannot be sure of the correctness of the system response. However, this task requires changes in business processes and is not considered in this work.

3.2. Business processes description

We consider a process of analyzing bank clients' card transactions for suspicion of fraud. Schematically, a bank payment's journey through the anti-fraud system can be represented as follows (Fig. 2).

- Bank clients who use card products via making online purchases, using pos-terminals, linking cards in order to conduct payments via smartphones, etc.
- Banking services for online payments, money transfers, and cash withdrawing.
- Bank anti-fraud system:
- Anti-fraud analytical platform, which is used to develop fraud detection algorithms. Banks with a large transaction flow exploit the Big Data technologies for these purposes.
- An engine to execute ML models online (model-based approach).
- Enrichment of transaction parameters with additional features created on the analytical platform.
- An engine to make a final decision on the operation based on expert rules (rule-based approach).
- Bank processing, in which an operation is performed once a verdict is returned by the fraud monitoring system.

In our case, the bank's fraud monitoring system combines two fraud detection engines: model-based (statistical analysis) that assesses risks using machine learning methods; and rule-based that assesses risks based on rules and the model-based risk scores that are used as the elements of the rules. The second engine is formed by fraud analysts as a result of fraud patterns analysis and is represented by a set of rules in a conjunctive normal form (CNF). The rules are checked sequentially at the time of the transaction ex-

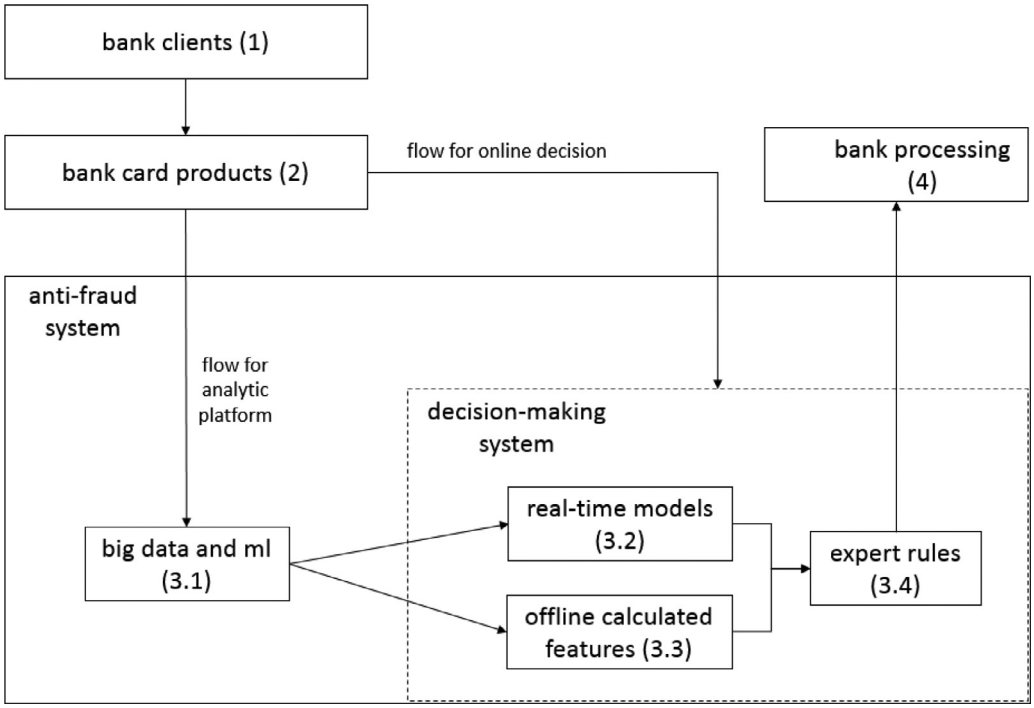


Fig. 2. One of the possible schemes of the bank's anti-fraud system, combining model-based and rule-based approaches.

ecution in the order of rules priority. For each rule, the action is defined – whether to block an operation if it satisfies the rule's conditions or "whitewash" it, excluding further checking of lower priority rules.

Thus, both the elements of artificial intelligence through which the new features for the anti-fraud system are created, and fraud analysts who directly adjust the rules, are involved in the process of transactional fraud monitoring. On the one hand, the combined approach allows the bank to adapt quickly to changes in fraudulent scenarios even if they changed just a few hours earlier. On the other hand, it facilitates developing the right strategy to train ML models, which require a large, labeled dataset and hence plenty of time to collect it.

The expert rules formation and adjustment are the regular processes which are essential to control the FBP and FPR metrics. Currently they are executed manually in the bank. Now, it is impossible to completely replace the expert approach with a model-based approach in a bank with high transactional activity. The underlying reasons for the manual control are as follows: a significant class imbalance while fitting the classification models, and a rapidly changing nature of fraud. These factors increase the likelihood of incorrect functioning of the antifraud system as long as the AI models deteriorate, which is reflected in the growth of rejected legitimate traffic or missed fraud. Also, fraudulent schemes are constantly changing and adapting, some of them lasting for a short period of time and covered effectively by simple rules. Accordingly, the classifier models, once having been fitted, are becoming less and less effective over the time, meanwhile updating and retraining them requires time and data science resources.

However, manual creation and modification of the rules that contain numerous scorings from the model-based engine makes it difficult to manage the system since it is getting more and more complex and interlinked. Much effort of fraud analysts is focused on reducing FBP, while business units of the bank require cybersecurity to ensure a positive customer experience characterized by the fraud monitoring not blocking legitimate transactions.

In this paper, we are concerned with the problem of model-based engine being biased towards the high recall of fraud detection. We propose to manage this issue by searching for rules that reduce false positives. The rules are formed by fitting decision trees and extracting the most effective features and conditions out of those created by analysts. The rules that are formed this way are highly interpretable and easily understood.

3.3. Data description

To conduct this research, we used cross-channel data on transactional fraud from one of the largest banks in Eastern Europe. The cross-channel nature of the anti-fraud system implies that various bank products, though differing in architecture, are connected to it. This approach allows accumulating data on events from different banking service channels (e.g., a mobile application, ATM, cards, SMS-banking, acquiring, bonus programs) into a unified analytical environment.

Based on the taxonomy of fraud (Onwubiko et al., 2020), we are concentrated on financial fraud committed through the online channels (Web, Mobile, Telephony) related to bank payments.

To produce rules, fraud analysts are provided with numerous attributes of operations (Table 1).

The response that the anti-fraud system returns as a result of a transaction evaluation includes:

- Predicted labels in the form of resolutions: 1 for the fraudulent and 0 for the legitimate transaction;
- Rules that are triggered on a transaction, and conditions and expressions that form the rules (Fig. 6).

4. Data processing and modeling

Our ML pipeline accounts for specificities of fraud detection:

- A strong imbalance of classes and unequal costs of types I and II errors;
- A need to quickly decide whether to block or allow transaction;

Table 1
Categories of data on card transactions.

Data category	Examples of attributes
Client profile	
	• Income and expenses • Geolocation • Consumer preferences
Merchant profile	
	• Merchant category code • Merchant turnover • Payment methods
Models-based estimates of risk score	
	• Merchant reliability • Client's propensity to carry out transactions in the particular merchant categories and banking service channels • Graph analytics of clients
Cross-product attributes	
	• Client profile in a bank mobile app • Credit scorings • Blacklists of merchants

- The lag between the time when transaction is conducted and the time when the final resolution on transaction is obtained.

We are concentrated on interpretable models, although they can be less precise than the black-box ones. We have chosen the tree-based classifiers – Decision Tree, Random Forest and Gradient Boosting. The branches of a tree constitute rules.

For quick decision-making, we create a model that is trained offline whereas the resulting rules are used online in a fraud-detection system. The model is re-trained on new samples and the rules are corrected on a regular basis. We do not have possibility to organize the process fully online with batch training due to the constraints dictated by our fraud-monitoring system (e.g., scenarios in which the monitoring logic could be organized).

The sample size bound us to utilize Big Data technology stack and distributed ML models.

The pipeline of our approach consists of three main stages:

- Data preparation and preprocessing;
- Modeling;
- Rules extraction and evaluation.

4.1. Data preparation and preprocessing

The first stage consists of the following steps (Fig. 3):

- Loading historical data from anti-fraud system;
- Anti-fraud system emulation;
- Dataset preprocessing: features selection, filtering out noisy data and extra features engineering.

The sample includes transactions over February 2021. Class 1 includes fraud missed and fraud stopped by antifraud system. Class 0 includes false positives, as well as the sampled legitimate transactions. We sample first minute of every hour, hence using systematic sampling with random starting point and a fixed periodic interval. This sampling method ensured data continuity and representativeness: based on our experience, the patterns of legitimate behavior change little over such period of time as two weeks, while legitimate operations of all possible types and patterns occur during the day due to a large flow of operations. For us, the random sampling performs worse in terms of the equal distribution of sample and population as indicated by Chi-Square and Kolmogorov-Smirnov tests.

On the emulation stage, we reproduce how the system would have worked on the historical data. Every transaction is checked for the correspondence of its parameters to the expressions and conditions used in expert rules. As a result, we obtain a set of boolean columns that correspond to expressions and conditions as well as the original features themselves that form the expert rules.

We preprocess dataset based on the experts knowledge of patterns in data, especially those that add noise and result in a low-quality ML models such as noisy labels, mistakes, repeated transactions of the same user relating to the same case, etc.

4.2. Modeling

During the second stage (Fig. 4), a prepared dataset is passed to the Decision Tree, Random Forest and Gradient Boosting classification models from the pyspark.ml library.[7]

Tree-based models are linear classifiers, but using conditions as features, expressions of which are joined on "OR", allows to add non-linearity to the models.

We tuned the models hyperparameters based on the k-fold cross-validation and customized confusion matrix as stated in (Höppner et al., 2020). It turned out that having so much data, it is hard to overfit the model. Also, the specific usage of model results through extracting rules rather than predicting class label does not allow to judge the classification quality of the rules based on the metrics corresponding to the decision tree which the rules come from. Thus, we believe that the computationally expensive and time-consuming grid search cross-validation step can be skipped. It is better to overfit and prune the rules in a cost-sensitive way.

Also, given such an use case, the ensemble models do not guarantee the better performance as we do not use their predicitions directly. Making ensembles out of rules should be based on sets of rules rather than on ensembles. In this work, we did not account for such possibility.

[7] https://spark.apache.org/docs/latest/api/python/_modules/pyspark/ml/classification.html

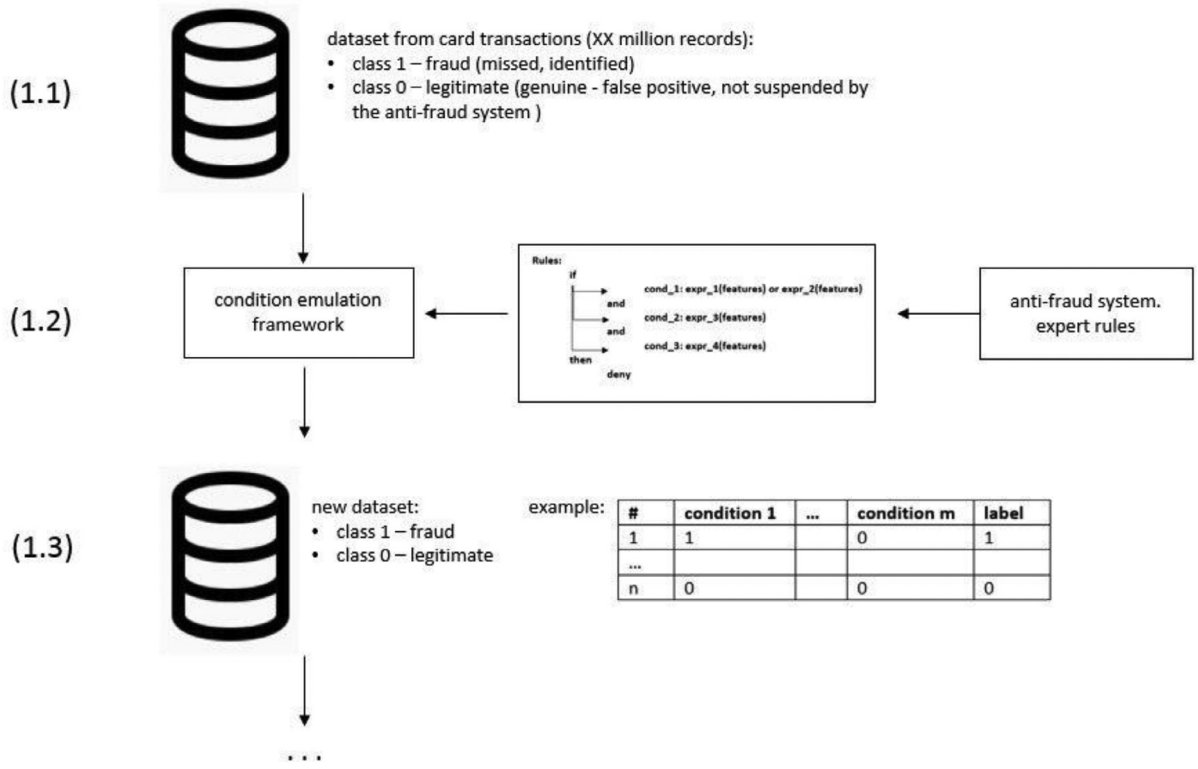


Fig. 3. Data preparation and preprocessing.

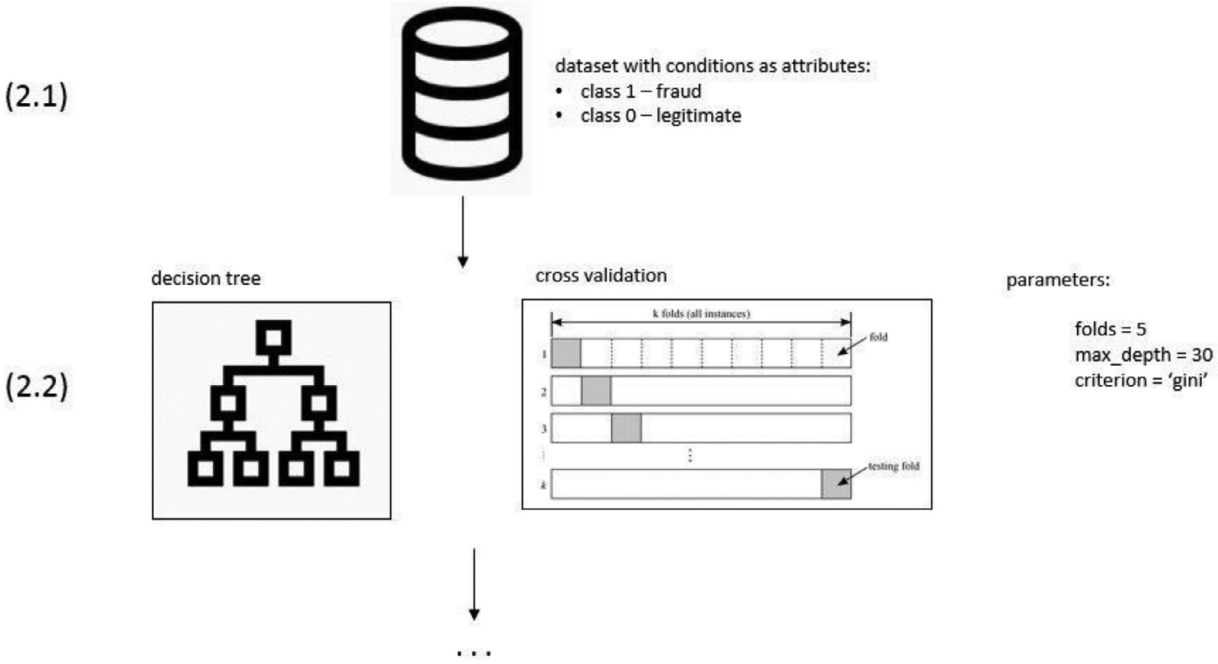


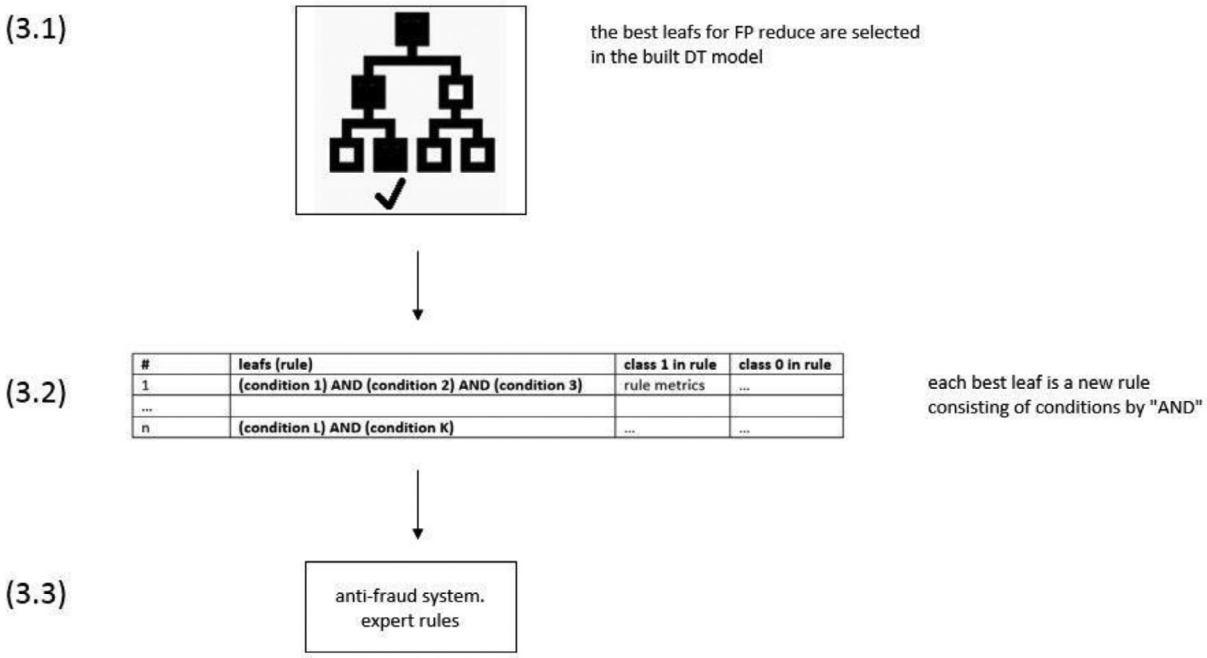
Fig. 4. Fitting the decision tree model and choosing its best specification.

4.3. Rules extraction and evaluation

The final stage is rules selection to incorporate them into a system (Fig. 5). It is made in three steps: 1) extracting rules from the fitted decision trees; 2) comparing rules with each other based on metrics with a given confusion matrix; 3) incorporating the best rules into the anti-fraud system so that they work in line with the expert rules.

The result of the decision trees fitting is new rules made out of the existing quantitative and categorical features, expressions, and conditions.

To extract specific rule branches from the tree, we used the graph representation of the tree (Kamiński et al., 2018). The rules constitute the sets of conditions and expressions that occur on the shortest path from the initial vertex to each of the terminal vertices.



#	leafs (rule)	class 1 in rule	class 0 in rule
1	(condition 1) AND (condition 2) AND (condition 3)	rule metrics	...
...			
n	(condition L) AND (condition K)	...	...

Fig. 5. Extracting rules from decision tree and choosing the best of them based on metrics.

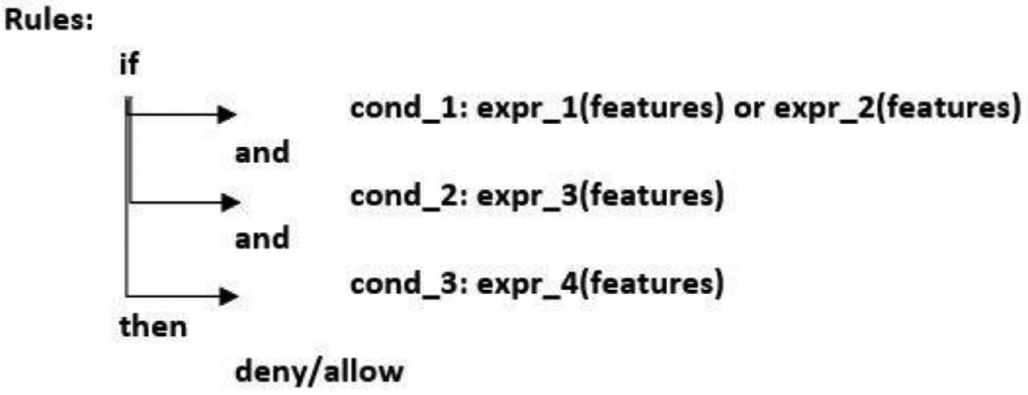


Fig. 6. The rule's structure.

After selecting the rules, we implemented a pruning technique. We tested every condition on how ML and business metrics change if it is removed from the rule. We chose to iteratively remove conditions one by one as doing it through subsets is a NP-complete problem.

Every rule is further converted to the CNF. Thus, we obtain the rules where conditions are joined on "AND" operator, whereas expressions within a condition are joined on "OR" operator (Fig. 6).

Both the rules predicting class 0 and rules predicting class 1 can be used to reduce false positives, though in different ways (Fig. 7). Every rule has an impact on transactions that are suspected by the system ("genuine" and "fraud_identified" areas in Fig. 7), as well as on transactions not suspected by the system ("fraud_missed" and "turnover" areas in Fig. 7). The main task is:

- For rules predicting class 0 – maximum coverage of false positives area (B, or "genuine") given the minimum coverage of fraud identified area (D);
- For rules predicting class 1 – maximum coverage of fraud missed area (C) given the minimum coverage of turnover area (A).

In accordance with the different ways in which two types of rules influence key business metrics, they are evaluated on the different sets of metrics. A rule for fraud detection reduces false positives if it replaces the less efficient existing rules with no increase in FP (Table 2). A rule for FP detection reduces false positives if it works over the existing rules, telling the system to allow transaction if the transaction seems to be the FP generated by the existing rules, with no increase in fraud missed (Table 3).

Due to the specificities in fraud detection that were discussed in Section 4.1, especially a time lag of fraud resolutions appearance, this type of metrics is suitable to evaluate rules on the held-out historical data. But if one needs to assess the rules performance online, she is doomed to wait for two weeks till major part of resolutions is here in order to calculate these metrics. We suggest a quick way to verify that the rule behaves like it is anticipated when training the model: we extrapolate the results obtained on the sample using the special formulas that one should construct based on the sampling strategy chosen. We rely on the anticipated number of rule triggers per minute. In our case, the formula to

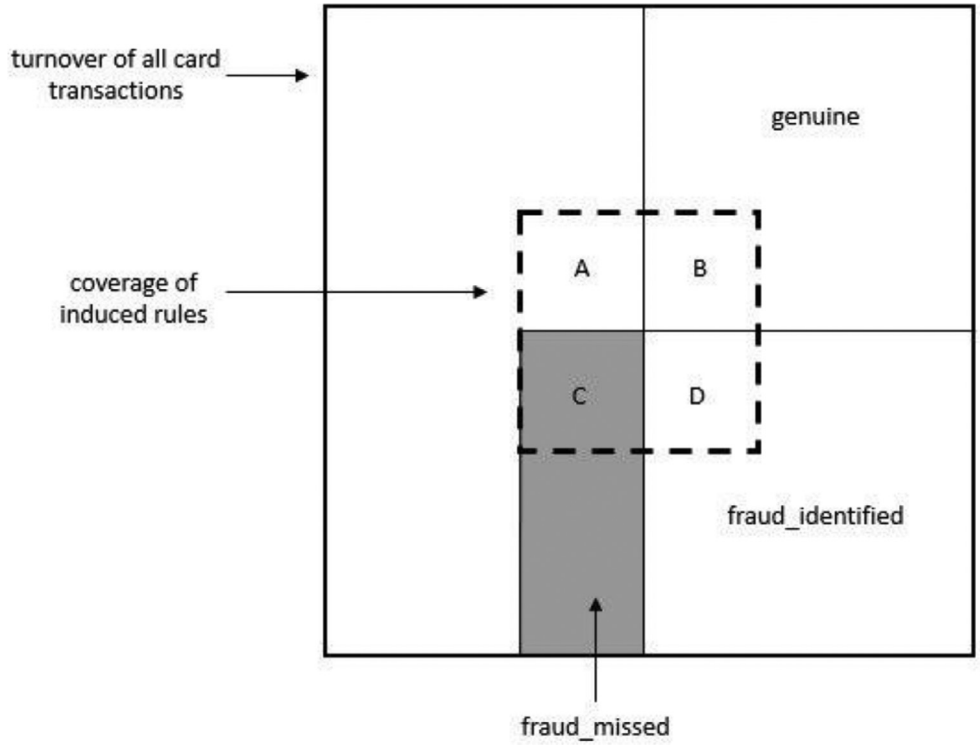


Fig. 7. The impact of the new rules on the transactions that pass through the antifraud system.

Table 2
Confusion matrix for fraud detecting rule.

Group	Predicted class	True label	Effect if the new rule increases the number or sum of the group
1	1	1 (fraud detected by existing rules)	No effect
2	1	1 (fraud missed)	Positive effect (plus sum of fraud missed)
3	1	0 (false positives)	No effect
4	1	0 (legitimate flow)	Negative effect (minus cost of contacting the client)

Table 3
Confusion matrix for FP detecting rule.

Group	Predicted class	True label	Effect if the new rule increases the number or sum of the group
1	0	1 (fraud detected by existing rules)	Negative effect (minus sum of transaction)
2	0	1 (fraud missed)	No effect
3	0	0 (false positives)	Positive effect (plus cost of contacting the client)
4	0	0 (legitimate flow)	No effect

conduct the extrapolations is the following:

$$Number\ of\ hits\ per\ minute = \frac{(\text{Triggers on legitimate flow})}{hours_1} + \frac{(\text{Other triggers})}{\text{hours} * \text{days} * 60\text{min}},$$

where *hours* is an average number of users' intensive activity hours per day (estimated by the bank experts), and *days* is the number of days in the sample.

The other metrics that can be used when deciding on which model or rule to choose include:

- The rule interpretability from the perspective of analytics and rule's correlations with expert rules;
- The simplicity of a rule measured by the number of its components and features;
- The universality of a rule measured by the number of cases that are regulated by that rule.

5. Results and discussion

The parameters of the fitted models are presented in Table 4. The Gradient Boosting model was the best of three.

In Table 5, we present the average metrics of 800 rules extracted from the decision trees and the bootstrap percentile confidence intervals.

The rules to put into the system were chosen based on the metrics introduced in Section 4.3:

- A rule to detect fraud: maximize the sum of the fraud, not stopped by the system but detected by the new rule, while minimizing the number of extra false positives generated by the rule;
- A rule to detect false positives: maximize the number of false positives generated by the system and detected by the rule, while minimizing the sum of the fraud stopped by the system, but misclassified by the rule as a false positive

Table 4
Models metrics.

model	dataset	precision	recall	fpr	accuracy	f1_measure	PR_AUC	ROC_AUC
Decision Tree	train	0.7463	0.8387	0.0091	0.9862	0.7898	0.7632	0.9674
Decision Tree	test	0.1581	0.7733	0.8160	0.2815	0.2625	0.2085	0.4753
Random Forest	train	0.7745	0.6156	0.0057	0.9825	0.6860	0.7459	0.9855
Random Forest	test	0.1422	0.5191	0.6205	0.4026	0.2232	0.1607	0.4424
Gradient Boosting	train	0.7732	0.8685	0.0082	0.9880	0.8181	0.8357	0.9919
Gradient Boosting	test	0.1615	0.8064	0.8292	0.2759	0.2691	0.2423	0.5470

Notes to the table: all the models were taken with max depth 10. Though the situation in terms of difference in metrics on train and test seems to be overfitting, it is still the best configuration of parameters.

Table 5
Bootstrap confidence intervals for rules.

Metric	Mean	Confidence interval
Precision	49.19%	(48.30%, 80.22%)
Recall	0.60%	(0.58%, 0.67%)
F1_measure	0.88%	(0.88%, 0.99%)

We experimented with a framework during half of the year. Each two weeks, we used to re-fit model using the latest data, generate new rules and put a few of the best to the anti-fraud system. The average precision of such a rule in online was approximately 10% while keeping recall at minimum level that provides the tangible business effect. The metrics depreciation in online mode is caused by inevitable differences of online compared to offline mode of anti-fraud system (e.g., the hierarchy of rules and other interdependencies between them in online).

It is worth mentioning that the approach requires much customization based on the application area, business goals and data specificities, and regular updating since fraud patterns change dynamically. Hence, we continue our experiments inducing rules on a regular basis and trying to improve on metrics and model robustness. Also, we are still working on reduction of the train, test and online metrics differences.

6. Conclusions and future research

This study confirmed the possibility and efficiency of utilizing the AI methods to recombine the conditions which form the expert rules used in the fraud monitoring systems. We proposed a Decision Tree approach and tested it on payments data derived from a large bank.

Our main goal was to reduce the number of false positives keeping the amount of missed fraud fixed at the current level. For this purpose, we suggested generating rules automatically, whether 1) those aimed at class 0 correcting the verdict of the fraud-detection system, and hence working over the current rules, or 2) those aimed at catching fraud with efficiency higher than the current rules efficiency and thus correcting (replacing) the original expert rules. Currently we got rules with average precision of 10% in online.

Our approach requires further improvement through customization and sample representativity check. However, as follows from the ongoing results, the approach has the potential to improve customer journey in banking and e-commerce. Furthermore, the effect can be achieved without adding new features or scores obtained by black-box classifiers to the anti-fraud system. Therefore, it guarantees no loss of interpretability of the verdict on the transaction. All in all, we consider it to be the antifraud analyst's support tool to induce new rules and reveal the most effective features and conditions for separating fraud.

We intend to develop our approach in several ways:

- Instance engineering: a more detailed study on sample representativeness and metrics extrapolation
- Cost-sensitivity: addressing the problem of unequal costs of classification mistakes directly through algorithm modification (e.g., adjusting impurity formula or using sample weights based on sum of transaction in models)
- Various ML methods for rule induction: fuzzy logic, non-linear non-tree-based algorithms
- Feature engineering: generating new features, including complex conditions
- Unsupervised learning: clustering clients before classifying their transactions
- Further automation of the rules induction process from the generation till the performance evaluation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Ivan Vorobyev: Conceptualization, Methodology, Writing – review & editing, Visualization, Validation, Supervision. **Anna Krivitskaya:** Data curation, Writing – original draft, Investigation, Software.

References

Baesens, B., Höppner, S., Ortner, I., Verdonck, T., 2021a. robROSE: a robust approach for dealing with imbalanced data in fraud detection. Stat. Methods Appl. doi:10.1007/s10260-021-00573-7.

Baesens, B., Höppner, S., Verdonck, T., 2021b. Data engineering for fraud detection. Decis. Support Syst. doi:10.1016/j.dss.2021.113492, 113492.

Baesens, B., Vlasselaer, V.Van., Verbeke, W, 2015. Fraud Analytics Using Descriptive. Predictive. and Social Network Techniques: A Guide to Data Science For Fraud Detection. John Wiley \& Sons. Inc, Hoboken. NJ. USA doi:10.1002/9781119146841.

Carrasco, R.S.M., Sicilia-Urbán, M.A., 2020. Evaluation of deep neural networks for reduction of credit card fraud alerts. IEEE Access 8, 186421–186432. doi:10.1109/ACCESS.2020.3026222.

Cirqueira, D., Helfert, M., & Bezbradica, M. (2021). Towards design principles for user-centric explainable AI in fraud detection doi:10.1007/978-3-030-77772-2_2

Hasanpour, Hesam & Ghavamizadeh, Ramak & Navi, Keivan. (2019). Improving rule based classification using harmony search. 10.7287/peerj.preprints.27634v1.

Höppner, S., Baesens, B., Verbeke, W., Verdonck, T., 2021. Instance-dependent cost-sensitive learning for detecting transfer fraud. Eur. J. Oper. Res. doi:10.1016/j.ejor.2021.05.028, S0377221721004562.

Hsin, Y.-., Dai, T.-., Ti, Y.-., Huang, M.-., 2021. Interpretable electronic transfer fraud detection with expert feature constructions. In: Paper presented at the CEUR Workshop Proceedings, p. 3052.

Kamiński, B., Jakubczyk, M., Szufel, P., 2018. A framework for sensitivity analysis of decision trees. Cent. Eur. J. Oper. Res. 26, 135–159. doi:10.1007/s10100-017-0479-6.

Leite, A., Gschwandtner, R., Miksch, T., Gstrein, S., Kuntner, J., 2020. NEVA: visual analytics to identify fraudulent networks. Comput. Graphics Forum 39 (6), 344–359. doi:10.1111/cgf.14042.

Onwubiko, C., 2020. Fraud matrix: a morphological and analysis-based classification and taxonomy of fraud. Comput. Secur. 96, 101900. doi:10.1016/j.cose.2020.101900.

Pant, P., Srivastava, P., 2021. Cost-sensitive model evaluation approach for financial fraud detection system. In: Paper presented at the Proceedings of the 2nd International Conference on Electronics and Sustainable Communication Systems, ICESC 2021, pp. 1606–1611. doi:10.1109/ICESC51422.2021.9532741.
Rao, Y., Ren, X., Duan, C., Mi, X., Cheng, J., Chen, Y., Wei, X. (2021). Knowledge-guided fraud detection using semi-supervised graph neural network doi:10.1007/978-3-030-90888-1_29
Sadgali, Imane, Sael, Nawal, Benabbou, Faouzia, 2021. Human behavior scoring in credit card fraud detection. IAES Int. J. Artif. Intell. 10, 698. doi:10.11591/ijai.v10.i3.pp698-706, IJ-AI.
Schneider, P. & Xhafa, F. (2022). Chapter 5 - Rule-based decision support systems for eHealth: supporting actors and stakeholders of health systems. Anomaly detection and complex event processing over IoT data streams, 87–99, doi: 10.1016/B978-0-12-823818-9.00015-8.
Severino, Matheus, Peng, Yaohao, 2021. Machine learning algorithms for fraud prediction in property insurance: empirical evidence using real-world microdata. Mach. Learn. Appl. 5, 100074. doi:10.1016/j.mlwa.2021.100074.
Sun, J., Li, Y., Chen, C., Lee, J., Liu, X., Zhang, Z., Xu, W., 2020. FDHelper: assist unsupervised fraud detection experts with interactive feature selection and evaluation. In: Paper presented at the Conference on Human Factors in Computing Systems - Proceedings doi:10.1145/3313831.3376140.
Trisanto, D., Rismawati, N., Mulya, M., Kurniadi, F., 2020. Effectiveness undersampling method and feature reduction in credit card fraud detection. Int. J. Intell. Eng. Syst. 13, 173–181. doi:10.22266/ijies2020.0430.17.
Sahin, Y., Bulkan, S., Duman, E., 2013. A cost-sensitive decision tree approach for fraud detection. Expert Syst. Appl. 40, 5916–5923. doi:10.1016/j.eswa.2013.05.021.
Sheng, V., Ling, Ch, 2006. Thresholding for making classifiers cos sensitive. In: Proceedings of the National Conference on Artificial Intelligence, 1.
Wedge R. Kanter J.M. Veeramachaneni K. Rubio S.M. Perez S.I. Solving the false positives problem in fraud prediction using automated feature engineering. In: Brefeld U. Curry E. Daly E. MacNamee B. Marascu A. Pinelli F. et al. editors. Mach. Learning Knowledge Discovery Databases. vol. 11053. Cham: Springer International Publishing; 2019. p. 372–88. https://doi.org/10.1007/978-3-030-10997-4_23.
Xie, Y., Liu, G., Cao, R., Li, Z., Yan, C., Jiang, C., 2019. A feature extraction method for credit card fraud detection. In: Paper presented at the Proceedings - 2019 2nd International Conference on Intelligent Autonomous Systems, ICoIAS 2019, pp. 70–75. doi:10.1109/ICoIAS.2019.00019.
Zeng, M., 2018. Smart business: What Alibaba's success Reveals About the Future of Strategy. Harvard Business Review Press, Boston. Massachusetts.
Zoldi, S., 2015. Using anti-fraud technology to improve the customer experience. Comput. Fraud Secur. 2015, 18–20. doi:10.1016/S1361-3723(15)30067-1.

Further reading

Carminati, M., Caron, R., Maggi, F., Epifani, I., Zanero, S., 2015. BankSealer: a decision support system for online banking fraud analysis and investigation. Comput. Secur. 53, 175–186. doi:10.1016/j.cose.2015.04.002.
Kumari P. Mishra S.P. Analysis of credit card fraud detection using fusion classifiers. In: Behera HS. Nayak J. Naik B. Abraham A. editors. Comput. Intelligence Data Mining. vol. 711. Singapore: Springer Singapore; 2019. p. 111–22. https://doi.org/10.1007/978-981-10-8055-5_11.
LiangJiejun, Hu Hu, Taihui, Li, Nannan, Xie, 2015. False positive elimination in intrusion detection based on clustering. In: 2015 12th International Conference Fuzzy Systems Knowledge Discovery. IEEE, Zhangjiajie. China, pp. 519–523. doi:10.1109/FSKD.2015.7381996.
Youssef, B., Bouchra, F., Brahim, O., 2020. Rules extraction and deep learning for e-commerce fraud detection. In: Paper presented at the Colloquium in Information Science and Technology, CIST,, 2020-June, pp. 145–150. doi:10.1109/CiSt49399.2021.9357066.
Zhou, H., Sun, G., Fu, S., Wang, L., Hu, J., Gao, Y., 2021. Internet financial fraud detection based on a distributed big data approach with Node2vec. IEEE Access 9, 43378–43386. doi:10.1109/ACCESS.2021.3062467.

Ivan A. Vorobyev. Graduated from Moscow State University of Instrument Engineering and Computer Science with a degree in Applied Mathematics (2006). In 2019, got an MBA degree at Sberbank Corporate University. In 2020, entered the postgraduate course on Information Security at the Higher School of Economics. The topic of primary research: "Machine learning and artificial intelligence methods for combating fraud in credit, finance and banking." Lecturer on anti-fraud technologies and artificial intelligence at Moscow State University, Higher School of Economics, Central Bank of the Russian Federation. Tutor at Higher School of Economics, with own course on tools for countering cyber fraud for students of Computer Security Department. Team Leader at Antifraud Acquiring in Sberbank of Russia. Twice (2020, 2021) received the prestigious VISA awards for "Lowest Gross Fraud - Acquirer" for the best indicators of anti-fraud in acquiring.

Anna D. Krivitskaya. In 2021, graduated from Lomonosov Moscow State University, BA's in economics. Currently doing master's degree on data analysis at Lomonosov Moscow State University. Working as a data scientist in cybersecurity department of Sberbank of Russia. Published scientific article on welfare economics at Journal of Economic theory (Russia).