Novel Machine-Learning-Based Decision Support System for Fraud Prevention

Norman Bereczki, Vilmos Simon, and Bernat Wiandt

1 Introduction

Over the past years, there has been a dramatic increase in the amount of data generated by people using the Internet. However, the development of hardware and software was significant, and scientists could not process a big amount of data with a personal computer. Dealing with huge amounts of data or doing calculations on complex problems became slow, and using supercomputers is not accessible for a wide range of projects. The intense growth of the amount of data and the increasing complexity of new computer technology solutions, such as blockchain, and the benefits of services gained interest for cloud computing.

Cloud computing is a service-based solution that allows the user to use services, such as IaaS, PaaS, SaaS, etc., rented from a provider. A major advantage of using cloud technologies is the better performance, the use of customizable hardware that does not require maintenance, and the better support for cooperative workflows. Based on these advantages, cloud service technology has gained large momentum in corporate environments [1]. Cloud services play a key role for a wide range of scientists and industrial processes.

Cloud platforms provide efficient resource allocation and several useful functions, such as creating automated backups. Registering on several major platforms is open for everyone. Thus it is unavoidable that fraudulent registrations will happen. Users are labeled as fraud, if somehow cause harm to the company: not paying their bills or engaging in illegal activities such as storing/streaming child pornography or mining cryptocurrencies.

N. Bereczki (✉) · V. Simon · B. Wiandt
Department of Networked Systems and Services, Budapest University of Technology and Economics, Budapest, Hungary
e-mail: norman.bereczki@edu.bme.hu; svilmos@hit.bme.hu; bwiandt@hit.bme.hu

A. Haldorai et al. (eds.), *5th EAI International Conference on Big Data Innovation for Sustainable Cognitive Computing*, EAI/Springer Innovations in Communication and Computing, https://doi.org/10.1007/978-3-031-28324-6_7

Table 1 Last 3-year complaint loss comparison in millions of $ [2]

Crime type	2021	2020	2019
Credit card fraud	173	130	111
Confidence fraud	956	600	475
Identity theft	278	219	160
Personal data breach	517	194	120

Table 1 shows that there is a massive increase in the loss because of fraudulent activities on the Internet.

Preventing malicious activity is important to protect the users of online services and to preserve their reputation. The majority of fraud cases can be detected during the registration process based on the recurring patterns coming from the registration information. Data is collected for each and every registration to the cloud platform. The data is mostly collected by the providers, but there are several third-party services providing information about the users. The data collected is used to determine whether a registered user is likely fraudulent or not. A user is considered anomalous, or called anomaly, if it is potentially going to do fraudulent activities. Currently, this work is done manually by analysts; thus, it is non-deterministic, because decisions made by people can be easily biased based on the subjectivity of people, such as experience or the current emotional state.

The aim of this research is to improve the efficiency of the fraud detection process by providing a decision support system for the analysts and implementing an unsupervised anomaly detection algorithm to validate the labeling. Every user that causes damage to the company (such as going against the laws or harming its reputation) is called fraud/fraudulent activity or anomaly.

The paper introduces a novel machine-learning-based approach that can be integrated into the existing decision-making framework performed by analysts. This new process eventuates a more deterministic way of classifying users and higher accuracy. The new method is also capable of giving feedback for already labeled data set thus improving the analysts' decision methodology. Our newly developed system relies on both supervised and unsupervised machine learning algorithms to provide multiple approaches to the problem.

The structure of this chapter is the following: Sect. 2 provides a brief summarizing about existing anomaly detection methods and fraud prevention systems. Section 3 details how the newly developed method works and what advantages does it have instead of using just the analysts' decision-making. Section 4 presents the evaluation of the introduced process on a real industrial data set and examines how it performs. Finally, Sect. 5 summarizes our methodology and its impact and proposes future development possibilities.

2 Related Works

Fraudulent activities are probably as old as humanity since people started using computers and telecommunication technologies so started criminals and scammers. The first amendment to the federal computer fraud law was enacted in the early stages of telecommunication and networks in 1986 [3]. Research into fraud detection due to its importance has a long history. The first discussions and analysis of computer fraud emerged during the 1970s from L. I. Krauss [4]. Fraudulent activities have been the subject of many studies in various fields. Since then, there has been an increasing amount of literature on preventing fraudulent activities [5].

There are a plenty of definitions of fraud. The Association of Certified Fraud Examiners defines fraud as "the use of one's occupation for personal enrichment through the deliberate misuse or misapplication of the employing organization's resources or assets" [6]. There are two types of fraud against companies: external and internal fraud. Internal fraudulent activities against a company is committed by an employee, for example, a sabotage. External fraud can be the activity of a user that harms the law, for example, not paying for the service, or streaming forbidden contents, such as related to terrorism. This chapter focuses on external fraudulent activities. Surveys such as that published by Abdallah et al. [7] have shown that the definition of fraud highly depends on the domain of the field it is observed on.

An efficient fraud prevention process includes a precise fraud detection. Fraud prevention is a critical aspect of online services because it has a high impact on the service provider's reputation. An online service can easily lose a large proportion of its customers if it has a reputation for being easily hackable [8].

A good approach to find fraud users is anomaly detection. Anomaly detection refers to the problem of finding patterns in data that do not conform to expected behavior. These nonconforming patterns are often referred to as anomalies or outliers [9]. The survey of Chandola et al. [9] provides a comprehensive review on anomaly detection methods and their usage. It discusses the variety of reasons how anomalies in data from several domains can be found [9]. Anomalous traffic patterns can indicate malicious attempts [9]. For example, anomalies in medical or healthcare data can mark abnormal patient conditions [9]. Anomalous user detection is often performed by graph-based fraud detection methodologies [10].

A large and growing body of literature has investigated user-profiling-based anomaly detection [11, 12]. The common purpose of this is to create user profiles and define distance thus creating clusters. The hypothesis what these algorithms are based on is that users in same clusters act same, so the goal is to identify fraud user clusters.

A common fraud activity is registering fictious accounts. Marakhtanov et al. proposed a long short-term memory (LSTM) recurrent neural-network-based methodology that can detect fraud users with 0.99 recall [13]. Sharma also investigates an LSTM autoencoder-based user behavior anomaly detection system that performs 0.91 recall [14].

Other neural networks than LSTM can perform well at anomaly detection tasks. In their study, Ding et al. show how neural-network-based technologies can be applied for anomalous user detection [15].

Hodge and Austin [16] identify 3 fundamental approaches to the problem of anomaly detection:

1. Determine the outliers with no prior knowledge of the data. A statistical approach is flagging the most outlying data in a given set based on statistical operations. Another approach is to perform unsupervised clustering with machine learning algorithms. Common algorithms from this field are k-nearest neighbor, connectivity-based outlier factor (COF) [17], one-class support vector machine [18], and neural-networks-based solutions such as the self-organizing map (SOM) [19] and the adaptive resonance theory (ART) [20].
2. Model the normal and abnormal behaviors. These methodologies require a labeled training set. The used methodologies for this approach are the scoring functions, the linear classifiers, the classification trees, and the nearest-neighbor methods [21].
3. Model normal behavior only. It is referred to a semi-supervised recognition task. Technologies utilized for this approach are the SSMBoost, the Boosting, the ASSEMBLE, and the SemiBoost [22].

A common approach of anomaly prevention in user space is to assign a profile to every user [23]. These profiles (often handled as matrix) contain information about predefined properties of a user. This profile can be compared with the target variable, and machine learning algorithms can detect correlation between the label and certain property values.

One of the earliest and most cited user-profile-based anomaly detection system has been presented by Lane et al. [24]. Their methodology presents a machine-learning-based approach to detect anomalous user activities, where the user profiles are collected from their UNIX typing sequences.

A popular machine-learning-based classifier is the random forest classifier. There are several papers about the successful use of the algorithm for anomaly detection [25–27].

Kater et al. present a study that uses state-of-the-art classification algorithms to filter out malicious users during registration [28]. They show that a supervised approach with the right features can provide a good basis for fraud prevention.

3 Machine-Learning-Based Fraud Prevention System

As Sect. 2 shows, there are a plenty of existing anomaly detection methods. A possible solution could be to implement a general solution, but the domain and the definition of anomaly differ highly in each case, thus more specific, domain-related models tend to have better performance, and moreover, they can be implemented to mimic the decision-making process of analysts more precisely. This section

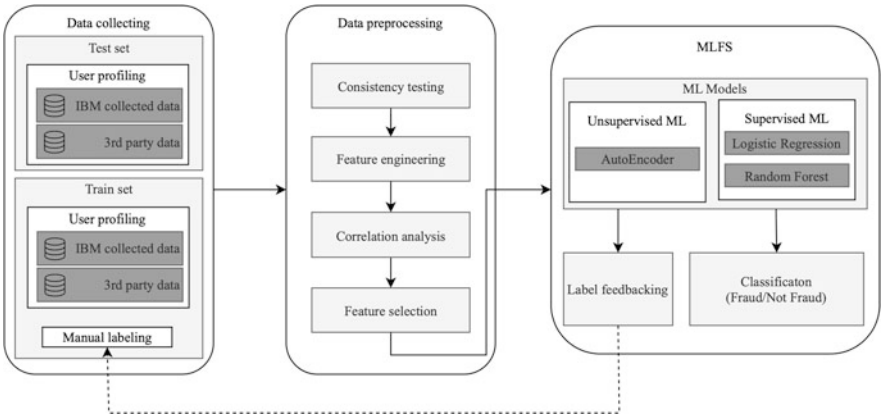

Fig. 1 Decision-making framework implementing MLFS

presents the machine-learning-based fraud prevention system (MLFS) developed by us, which uses the exact same features that are used by analysts for decision-making. This newly introduced system handles numerical and categorical features of a registering user as an input and returns a probability that shows how likely is it that the user is fraud. MLFS is integrated into the decision-making process. By making a decision, the analysts take the suggestion made by MLFS into consideration to decide whether a user is labeled as fraud or not. Use of MLFS leads to a more deterministic process, because it is based on exact mathematical models, while human decision-making can be highly influenced by the personal elements, such as the experience level or current emotional status.

Figure 1 shows how the decision-making framework implementing MLFS builds up. It consists of 3 main components: data collecting, data preprocessing, and MLFS. The data collection part collects the user profiles and assigns labels to the training data points. Data preprocessing block is responsible for testing the consistency of the train and test sets, generating new features, and performing correlation analysis. The goal of this block is to enable the process to rely on features that have high descriptive power. After the input is prepared, MLFS is ready to be utilized. Both supervised and unsupervised approaches are implemented and contribute to detect anomalies. Supervised algorithms are trained on the pre-labeled data by analysts. Basically, the supervised approach models and mimics the decision-making of analysts. This indicates a logical limit because if the classification algorithm reaches up to perfect accuracy, it is still only as accurate as the analysts' labeling. There might be misclassified cases in the training data set.

To improve the performance and eliminate this limitation of fraud prevention, an unsupervised approach is also implemented. It determines the outliers based on mathematical operations without labels. The final classification process of a new user is shown in Fig. 2. The pre-processed user profile is passed to the analyst and to the pre-trained ML models. The supervised and unsupervised approaches calculate

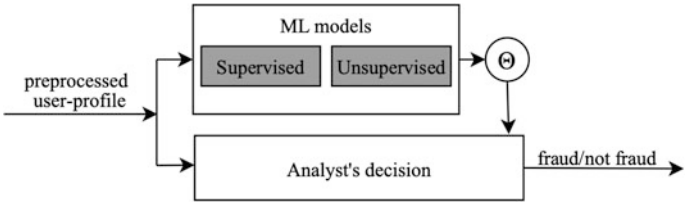

Fig. 2 Decision-making process with MLFS

the probability of fraud case, and then Θ calculates the arithmetic mean of the two probabilities and passes it to the analyst. The analyst's decision is highly supported by the implemented machine learning algorithms.

For the supervised approach, the logistic regression (LR) and the random forest classifier (RF) are used. These algorithms have been selected based on several reasons. Both are lightweight algorithms that do not need big computational capacity. Both algorithms are easy to understand and do not operate black-box-like as most of the neural-network-based classifiers do. LR and RF can also prevent overfitting. These algorithms are commonly used and well-studied ML algorithms. These two algorithms are really accurate when performing one-class classification, LR performs better in overall accuracy, but the true positive rate and false positive rate are higher for RF with increased noise variables [29]. By using deep neural-network-based classification solutions, it could potentially improve the effectiveness of the classification; however, the number of available labeled data set is too small to train a supervised neural network model.

LR is one of the mostly used statistical models to calculate the probability of one event based on the linear combination of more independent variables. These variables, called predictors, are the normalized values in a user profile. The output of LR is a continuous probability variable bounded between 0 and 1 that shows that the probability of a newly registering user will be fraud. To prevent overfitting, a cross-validation set is used, that is, the 5% of the test set.

RF is a robust, very popular classification method. It initializes several independent decision trees and uses the most voted results by the decision trees. Decision trees are decision support systems that represent the final result of successive decision sequences. These decision can be based on probabilities, cost functions, etc. RF is a popular algorithm because it is easy to use, has very high accuracy, and provides solution for overfitting by using bagging. It uses just a subset of the data set during teaching and performs validation with the other data points. It makes unnecessary to perform cross-validation, since this operation corresponds exactly to that.

The unsupervised part of MLFS implements an **autoencoder neural network**.

As discussed earlier in this section, our goal is to implement an anomaly detection process that is independent from the labels made by analysts thus avoiding the logical limitation. Autoencoder is a generative unsupervised neural network that is commonly used for anomaly detection. It consists of two main parts: encoder

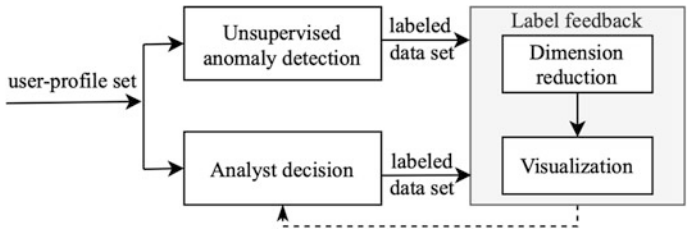

Fig. 3 Feedback to the analysts' decision-making with MLFS

and decoder. The encoder takes a user profile X as an input and compresses the given data into lower-dimensional latent subspace z. After this, the decoder takes the compressed data z and the decoder reconstructs the original data X'. Then X and X' are compared, and the reconstruction error (mean square error) is calculated. It is assumed that the parameters of a fraudulent user are different from a normal user. The autoencoder learns how to compress and decompress data points that represent users with normal behavior. If an anomalous case occurs, the decompression part can only reconstruct it with high error. If the error crosses the threshold, the case is marked as anomalous.

The MLFS can also re-classify an existing labeled user profile set based on the unsupervised (autoencoder) approach shown in Fig. 3. The unsupervised anomaly detection is performed on the same database that is labeled by the analysts. The user profiles, labeled by both the unsupervised algorithm and analysts, are then transformed into a 3-dimensional latent space to be able to visualize it. In this latent space, the results of the two decision-making methods are compared. This comparison may show cases that are marked as anomalous by the unsupervised learning but not by analysts. These cases are then marked and sent back to the analysts for a deeper analysis. This process gives a feedback for the analysts' decision-making. If the autoencoder implementation can correct the imperfection of the labeling process that can lead to a labeled training data set with less fault, thus the quality of the training data set improves. A better training set results in more accurate supervised classification. The cooperative use of both supervised and unsupervised approaches leads to a self-improving system.

To be able to visually compare the labels given by the analysts and by the autoencoder, dimension reduction is implemented to overcome the visualization difficulty of user space by its high dimensionality. Dimension reduction is a process that reduces data from a high-dimensional space to a lower-dimensional space with minimal loss of information using mathematical operations (e.g., different projections).

MLFS uses the **UMAP** (Uniform Manifold Approximation and Projection for Dimension Reduction) dimension reduction method. UMAP has several advantages over the popular dimension reduction method t-SNE, but the two most significant are the radical reduction of the running time of dimensional reduction on large sets and the better preservation of local data structures that is an important part

of visual cluster analysis for anomaly detection [30]. MLFS reduces the data into a 3-dimensional latent space and plots each compressed user profile as a data point, where the colors of the points indicate the label of the anomaly detection process. Then the labeling of the autoencoder and the analysts can be visually compared. A big difference between the two labeling indicates that the autoencoder-based method is not able to perform anomaly detection properly, because much of the analysts' labeling can be considered correct. If the comparison shows a slight difference, then the differing users are sent back to the analysts for more observation, because the ground of the difference can be that the analysts did not recognize the anomalous pattern properly.

In summary, MLFS can contribute to the decision-making by:

1. Proposing a classification methodology based on various machine learning approaches
2. Using a mathematical way to decide whether a new registering user is anomalous or not thus eliminating the human intuition and making the process more deterministic
3. Giving feedback for the labeled data set thus improving the labeling mechanism used by analysts

4 Experiments and Results

4.1 Data Collecting

MLFS uses existing data to determine whether a new user is potentially fraud user or not. It is collected by IBM and third-party services, which are to provide information about the context of a previous occurrence of a user, based on its username or e-mail address.

The data for this study was generated by IBM Budapest Lab for educational purposes. The used databases are synthetic and GDPR compliant, so it is not possible to determine how much of the world of the IBM Cloud they cover or when the data was recorded or how much it reflects the current set of users. This data set was labeled by analysts. The anomalously marked cases were pretty rare, and they accounted for only 20% of the data set making it the data set unbalanced by the two categories (fraud/not fraud). The data consists of 500.000 records with about 300 features. There is also an unlabeled data set available during the project that has not been labeled manually yet containing 2.000.000 records.

For features to be passed to a model, a unified structure is required. Because the data came from multiple sources, their occupancy rates were different, so features that were missing by more than 10% of the entities from the labeled data set were dropped. The categorical variables have been re-coded into a maximum of 4 categories, keeping the most significant 3 and one "other" category. Approximately, 30 properties remained.

4.2 Data Preprocessing

To use the labeled data set for teaching the model that will be evaluated on the unlabeled data set, it is important that the features in the two data sets are consistent with one another. Consistency means that the sets of values for the same features in the two databases are statistically identical. Using consistent training and validation data set is essential for good predictions. The machine learning algorithm learns the patterns from the training data set. In case when the validation data set is not similar, the characteristic of these patterns can differ that can intensely bring down the efficiency of the algorithm, because it is trained to recognize patterns in the training set. To determine whether the labeled and unlabeled data sets are similar, the statistical analysis of variables has been performed.

For continuous variables, the goal is to disprove the null hypothesis. The null hypothesis claims that there are no statistical relationship between two sets. The null hypothesis persists until proven otherwise [31]. The alternative hypothesis claims that the values of the two variables are statistically related, making the two data sets similar. To prove this, the *paired t-test* and the *Smirnov test* are used. The t-test examines whether the mean of two variables differs statistically [32]. The Smirnov test is a two-sample version of the Kolmogorov–Smirnov test, which is designed to show if two samples are identically distributed. The two samples are statistically similar if the p-value of the test is less than 0.05. Both of the statistical tests showed that the labeled and unlabeled data sets are consistent through continuous features.

To examine categorical variables, the distribution of the occurrence of unique values has been tested. The distribution of every categorical feature has been compared. The comparison showed no significant difference between the occurrence of each value in the two data sets thorough the same feature. However, when examining the categorical variables, it was found that the labeled set had a greater fill rate. This may be due to the fact that users in the labeled data set use a paid service, and these users are more likely to provide more data, against those users that will use just the free plan services.

MLFS only performs operations on numerical data, so it is important to convert the categorical text features into numerical. The *one-hot encoding* was used for the conversion. For a feature with n category, one-hot encoding generates an $n - 1$ high vector. The trivial base vectors (unit vectors) and the null vector represent each category. This has the advantage over label encoding, which assigns an integer to each category, to define the same distances between categories.

From the existing properties, new properties are created that can increase the accuracy of the method. The location of the IP address was available from various sources, where the billing takes place, and whether the user is using a VPN or other servers that mask the IP. If this is not used, but the IP address and the billing address do not match, a Location Unsimilar bit is set to 1, indicating that the two locations are different. Otherwise, its value is 0. If there is missing data, its value is also set to 1 as a precaution.

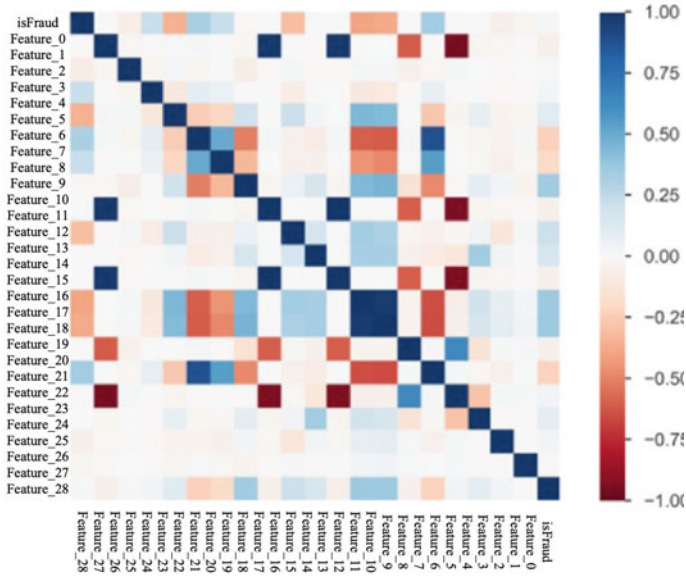

Fig. 4 Pearson's correlation matrix

In order to select the features with high descriptive power for the model, two correlation coefficients are observed. **Pearson's correlation coefficient:** It is a frequently used correlation metric that can detect the linear association between two variables. Its coefficient range is from -1 to 1. If the r coefficient is positive, it represents a positive relationship between the variables; otherwise, if negative, it represents a negative correlation. ϕ_k **correlation coefficient:** The ϕ_k correlation coefficient has several advantages. It can be used between categorical, continuous, and interval variables. Unlike Pearson's r, it is also capable of detecting nonlinear relationships. If the two variables are Gauss-distributed, it leads the problem back to the Pearson r coefficient; otherwise, it calculates the χ^2 test.

The Pearson's correlation coefficient and the ϕ_k correlation coefficient showed identical results. Figure 4 shows the Pearson correlation matrix of the features and the top 10 features have been selected to the model.

4.3 Performance Evaluation of MLFS

To measure the accuracy of MLFS classification, accuracy, precision, recall, and F1-score metrics are used (TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative):

– Accuracy:

$$\frac{TP + TN}{TP + TN + FP + FN}. \tag{1}$$

– Precision:

$$\frac{TP}{TP + FP}. \tag{2}$$

– Recall:

$$\frac{TP}{TP + FN}. \tag{3}$$

– F1-Score:

$$\frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2 * TP}{2 * TP + FP + FN}. \tag{4}$$

Precision shows how many of the anomalously classified users are truly anomalous. In contrast, recall shows how many anomalous users have been labeled as anomalous out of every anomalous user from the user space. If the algorithm retrieves 60 anomalies out of 100 anomalous cases and marks another 60 users as anomaly that are indeed not, that means 0.5 precision and 0.6 recall. But if the algorithm retrieves 20 anomalies out of 100 truly anomalous cases and retrieves 0 false positive cases, that means 1 precision but only 0.2 recall. It is important to know these metrics, so the MLFS can be optimized for the right usage. The goal of this system is to optimize it to find as many truly anomalous cases as possible instead of optimizing it to label as few normal cases as anomalous as possible.

Table 2 shows the performance measurements of MLFS. As Table 2 shows, RF reached better performances in every aspect than LR; however, the difference is not significant. The results are adequate, and the developed system can recommend labels with high accuracy.

Autoencoder was trained on the labeled data set without passing it the labels. The trained autoencoder model was evaluated by comparing the predicted labels with the output. Figure 5 shows the visualized comparison of the autoencoder labeling and the analysts labeling.

Table 2 Performance measurements of classification models

	Accuracy	Precision	Recall	F1-score
Logistic regression	0.800	0.741	0.664	0.684
Random forest classifier	0.806	0.750	0.672	0.693

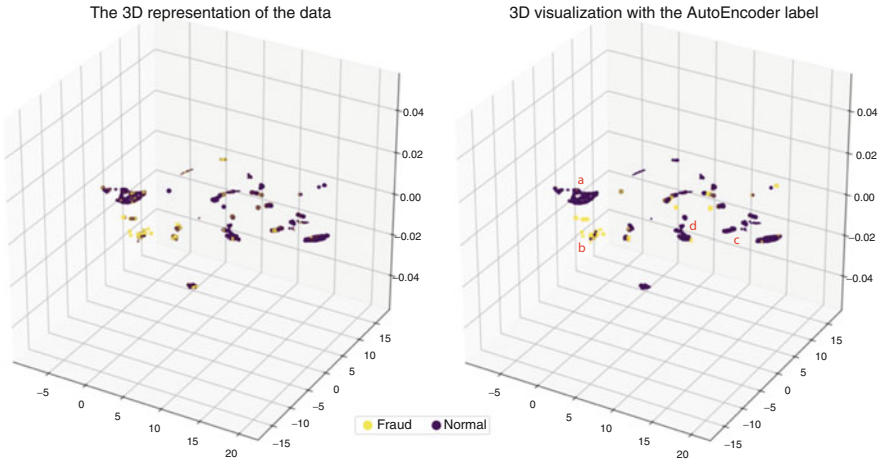

Fig. 5 Comparison of labels given by the analysts and the autoencoder

Figure 5 shows that no significant difference can be seen between the labels (left) and the predicted labels (right). This means that the fraud prevention can be successfully solved by unsupervised models that do not rely on domain knowledge to label the data. Normal behavior user grouping (groups a, b, and c) and anomalous grouping (group b) occur in the user space; however, outliers can be found in these groups. Cases classified differently by the autoencoder have been sent back for revision to the analysts to determine whether the classification was correct or not. The results are as expected, autoencoder marked largely the same users fraud, and however, not every label is the same. This unsupervised approach of MLFS can indicate potentially mislabeled users.

5 Conclusions

A novel decision support system has been created that can predict with great correctness whether a new user is fraud or not. MLFS uses a logistic regression and random forest classifier to implement a supervised classification-based anomaly detection that relies on the analysts' labeling. RF overperformed LR in every metrics, but the difference was not significant. The MLFS can reach 0.8 accuracy, 0.75 precision, and 0.67 recall on the IBM Cloud Prepared data set. A recommendation system has been proposed that provides a strong support for the analysts. Later on, it can also replace human work, resulting in a much faster registration process, because a new user does not have to wait for the analysts to examine its case.

The implementation of the proposed MLFS system makes the user fraud prevention more deterministic, the human factors of the analysts will not have such an impact on the decision, so the filtering process can be much more consistent.

This system has brought into focus features for the analyst that have not been observed with great emphasis; however, the autoencoder and the correlation analysis showed that they have a great impact on the label. The results obtained during the feature and correlation analysis were demonstrated several times to the analyst teams. Their attention was drawn to several features that showed a high correlation with the label, but so far it has not been observed with great attention, thereby improving the accuracy of their work, and thus even more users who harm IBM Cloud and the company can be filtered out in advance. This information has been implemented into the actual decision-making system.

The developed process can provide feedback to the classification. The decision-making of the analysts can be supervised thus improving its effectiveness. This can improve the quality of the labeling, resulting in more accurate models.

Future possibility is to try out additional machine learning or deep learning models to increase the performance of the novel process. For this, it is necessary to collect more data, both labeled and unlabeled ones.

Acknowledgments The research reported in this paper is part of project no. BME-NVA-02, implemented with the support provided by the Ministry of Innovation and Technology of Hungary from the National Research, Development and Innovation Fund, financed under the TKP2021 funding scheme.

References

1. P. Koehler, A. Anandasivam, D. Ma, Cloud services from a consumer perspective, in *Proceedings of the 16th Americas Conference on Information Systems (AMCIS 2010), Lima, Peru* (2010)
2. Internet Crime Complaint Centre IC3, *Internet Crime Report 2021*. Federal Bureau of Investigation (2021)
3. D.S. Griffith, The Computer Fraud and Abuse Act of 1986: a measured response to a growing problem. Vand. L. Rev. **43**, 453 (1990)
4. L.I. Krauss, A. MacGahan, *Computer Fraud and Countermeasures* (Prentice-Hall, Englewood Cliffs, 1979)
5. A.A.Z. Mansour, A. Ahmi, O.M.J. Popoola, A. Znaimat, Discovering the global landscape of fraud detection studies: a bibliometric review. J. Financial Crime **29**(2), 701–720 (2022)
6. Association of Certified Fraud Examiners, *Report to the nations on occupational fraud and abuse*. Association of Certified Fraud Examiners (2002)
7. A. Abdallah, M.A. Maarof, A. Zainal, Fraud detection system: a survey. J. Netw. Comput. Appl. **68**, 90–113 (2016)
8. A.O. Hoffmann, C. Birnbrich, The impact of fraud prevention on bank-customer relationships: an empirical investigation in retail banking. Int. J. Bank Marketing **30**, 390–407 (2012)
9. V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: a survey. ACM Comput. Surv. **41**(3), 1–58 (2009)
10. T. Pourhabibi, K.-L. Ong, B.H. Kam, Y.L. Boo, Fraud detection: a systematic literature review of graph-based anomaly detection approaches. Decision Support Syst. **133**, 113303 (2020)
11. M. Chen, A.A. Ghorbani, et al., A survey on user profiling model for anomaly detection in cyberspace. J. Cyber Security Mob. **8**(1), 75–112 (2019)

12. R. Ramachandran, R. Nidhin, P. Shogil, Anomaly detection in role administered relational databases—a novel method, in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (IEEE, Piscataway, 2018), pp. 1017–1021
13. A.G. Marakhtanov, E.O. Parenchenkov, N.V. Smirnov, Detection of fictitious accounts registration, in *2021 International Russian Automation Conference (RusAutoCon)* (IEEE, Piscataway, 2021), pp. 226–230
14. B. Sharma, P. Pokharel, B. Joshi, User behavior analytics for anomaly detection using LSTM autoencoder-insider threat detection, in *Proceedings of the 11th International Conference on Advances in Information Technology* (2020), pp. 1–9
15. Z. Ding, L. Liu, D. Yu, S. Huang, H. Zhang, K. Liu, Detection of anomaly user behaviors based on deep neural networks, in *2021 IEEE 20th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)* (IEEE, Piscataway, 2021), pp. 1240–1245
16. V. Hodge, J. Austin, A survey of outlier detection methodologies. Artif. Intell. Rev. **22**(2), 85–126 (2004)
17. O. Alghushairy, R. Alsini, T. Soule, X. Ma, A review of local outlier factor algorithms for outlier detection in big data streams. Big Data Cognit. Comput. **5**(1), 1 (2020)
18. M. Goldstein, S. Uchida, A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data. PLOS ONE **11**, 1–31, 04 (2016)
19. X. Qu, L. Yang, K. Guo, L. Ma, M. Sun, M. Ke, M. Li, A survey on the development of self-organizing maps for unsupervised intrusion detection. Mob. Netw. Appl. **26**(2), 808–829 (2021)
20. S. Omar, A. Ngadi, H.H. Jebur, Machine learning techniques for anomaly detection: an overview. Int. J. Comput. Appl. **79**(2), 33–41 (2013)
21. E. Carrizosa, D.R. Morales, Supervised classification and mathematical optimization. Comput. Oper. Res. **40**(1), 150–165 (2013)
22. J.E. Van Engelen, H.H. Hoos, A survey on semi-supervised learning. Mach. Learn. **109**(2), 373–440 (2020)
23. C.S. Hilas, J.N. Sahalos, User profiling for fraud detection in telecommunication networks, in *5th International Conference on Technology and Automation* (2005), pp. 382–387
24. T. Lane, C.E. Brodley, An application of machine learning to anomaly detection, in *Proceedings of the 20th National Information Systems Security Conference*, vol. 377, Baltimore, USA (1997), pp. 366–380
25. R. Primartha, B.A. Tama, Anomaly detection using random forest: A performance revisited, in *2017 International Conference on Data and Software Engineering (ICoDSE)* (IEEE, Piscataway, 2017), pp. 1–6
26. J. Zhang, M. Zulkernine, A. Haque, Random-forests-based network intrusion detection systems. IEEE Trans. Syst. Man Cyber. Part C (Appl. Rev.) **38**(5), 649–659 (2008)
27. M.A.M. Hasan, M. Nasser, B. Pal, S. Ahmad, Support vector machine and random forest modeling for intrusion detection system (IDS). J. Intell. Learn. Syst. Appl. **2014**, 45–52 (2014)
28. C. Kater, R. Jäschke, You shall not pass: detecting malicious users at registration time, in *Proceedings of the 1st International Workshop on Online Safety, Trust and Fraud Prevention* (2016), pp. 1–6
29. K. Kirasich, T. Smith, B. Sadler, Random forest vs logistic regression: binary classification for heterogeneous datasets. SMU Data Sci. Rev. **1**(3), 9 (2018)
30. L. McInnes, J. Healy, J. Melville, UMAP: Uniform manifold approximation and projection for dimension reduction (2018). Preprint arXiv:1802.03426
31. D.R. Anderson, K.P. Burnham, W.L. Thompson, Null hypothesis testing: problems, prevalence, and an alternative. J. Wildlife Manag. **64**, 912–923 (2000)
32. T.K. Kim, T test as a parametric statistic. Korean J. Anesthesiol. **68**(6), 540–546 (2015)