ISBN 978-981-14-1684-2
Proceedings of 2019 the 9th International Workshop on Computer Science and Engineering
(WCSE 2019 SUMMER)
Hong Kong, *15-17 June*, *2019*, *pp. 551-557*
doi: 10.18178/wcse.2019.06.082

Abnormal Detection of User Behavior in Online Banking

Yuan Wang [1,2], Liming Wang [1+], Wei An [1]

1 Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
2 School of Cyber Security, University of Chinese Academy of Sciences
wangyuan@iie.ac.cn, wangliming@iie.ac.cn, anwei@iie.ac.cn

Abstract. Abnormal detection is very important in online banking security. One of the most difficult issues in abnormal detection is how to calculate the distance between data samples. After the analysis of user behavior of online banking, we propose a mixed method based on Euclidean distance and cosine similarity, to measure the similarity among user behaviors. This paper develops an approach to catch similar behaviors to the abnormal behaviors in online banking transactions, by using the mixed similarity measurement. Experiment results show that our method can improve the performance of abnormal detection on the underground dataset, comparing to Euclidean distance and cosine similarity.

Keywords: Online banking; user behavior; abnormal detection

1. Introduction

The popularity of online banking has increased year by year in China as reported by CFCA (China Financial Certification Authority). At the same time, a variety of security issues are growing.

As discussed in [1], one of the most difficult issues researcher faces is because of the limited availability of malicious samples (owing to obvious privacy concerns). Therefore, abnormal detection is of the essence to resolve security issues of online banking. However, there are remaining issues of abnormal detection as described in [1], one of which is how to calculate the distance between data samples.

Thanks to the cooperation with a Chinese bank, who offered us the transaction data of online banking system, we have an opportunity to detect abnormal behaviors in the transaction data. We analyze the transaction data and provide the bank some extreme events which seem abnormal. The bank acknowledges only a few bad transactions because of limited manpower and material resources. Then we try to find the similar bad behaviors based on the limited seeds in all transactions. The challenge we face is to find a distance measurement method that suits our requirements. In this paper, distance is a quantitative metric for deviation of user activity sequence. One activity sequence represents a user behavior from login to logout.

To find the bad behaviors that similar to the seeds, we adopt a framework of clustering approach, which divide the normal behaviors and abnormal behaviors. We leverage the Louvain Clustering method to form communities, then spot bad communities which include abnormal seeds. Our contributions are following:

1) We propose a similarity measurement of mixed distances to support the clustering method, and it works better than present methods.

2) We develop an algorithm of abnormal detection based on Louvain Clustering method, which is a very efficient community detection method.

3) We practice our method on a real-world dataset, and show that our method performs well to catch the similar bad behaviors in online banking transactions.

+ Corresponding author. Tel.: 13693560895
E-mail address: wangliming@iie.ac.cn

The reminder of this paper is organized as follows: In Section 2, we describe the data set. Section 3 introduces our detection approach. In Section 4, we investigate the performance of our detection method on ground truth data. Section 5 discusses the related work, while Section 6 concludes this paper.

2. Data Description

With the online banking system, customers log in to access the online banking service through a browser, or access the online banking service through an account marked with an individual customer or company customer, and they can perform various financial activities, various inquiries, money transfers, fee payments, investments. Each request of a customer will be stored as a record in the data.

Our data is collected from one of the top banks in China's online banking system which provides online services for millions of customers every day. The data contains various information about the customer's behavior, such as time stamp, account ID, payer ID, payee ID, operation (e.g. login, money transfer), amount, login IP, login area, operation status (success or failure).

A session is a period of a customer's activity from its login to its logout with a session ID. Our data includes 101,833,505 records and 23,852,308 sessions in total. After filtering the error records, we finally have 23,212,800 sessions and 91,002,483 records. If there is at least one transaction (money-moving operation) in the session, we call the session a transaction session; otherwise, we call it a non-transaction session.

3. Detection Method

Because of the lack of large labeled samples, we promote a framework to find abnormal behaviors based on clustering algorithm. We analyze the user click behavior and build a similarity graph of user click sequence, and then we divide the graph into clusters. Finally, we spot bad clusters which including abnormal seeds.

3.1 Click activity

To gain a holistic view of user activities in online-bank, we use the clickstream model\cite{b2}. We build a first order Markov chain of user activities and compute the probability transition between every pair of activity states. As shown in Figures 1, nodes represent actions users take and directed edges represent the transition between two actions.

We add two abstract states, INITIAL and FINAL, because of that some sessions do not begin with login and end with logout. Although the sum of outgoing transitions from each state is 1.0, we only keep the edges with probability > 0.05 in the following figures for complexity. In fact, there are thousands of states in the online banking system, as so many financial services available.

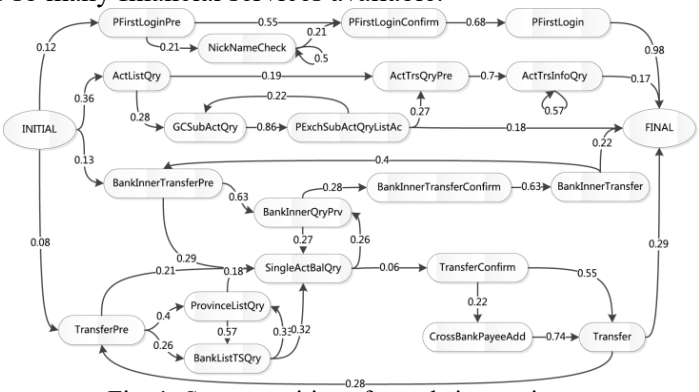


Fig. 1: State transitions for website sessions.

After pruning edges, we can finally derive four main activity sequences:

1) The first-time login activity seq_1^w (*INITIAL, PFirstLoginPre, PFirstLoginConfirm, PFirstLogin, FINAL*), describes how users complete their first login with some checks like nickname check.

2) The query activity seq_2^w (*INITIAL, ActListQry, GCSubActQry, PExchSubActQryListAc, ActTrsQryPre, ActTrsInfoQry, FINAL*), shows the common path for queries of account balances or recent transaction information. It is the most frequent transactions in the online banking system.

3) The innerbank transfer activity seq_3^w (*INITIAL, BankInnerTransferPre, BankInnerQryPrv, BankInnerTransferConfirm, BankInnerTransfer, FINAL*), depicts how users transfer their money to accounts in this online banking system.

4) The interbank transfer activity seq_4^w (*INITIAL, TransferPre, ProvinceListQry, BankListTSQry, SingleActBalQry, TransferConfirm, Transfer, FINAL*), describes how to transfer funds to accounts out of the online banking system.

In Figure 1, we can find that to complete one activity, users mainly take 4 (seq_1^w), 6 (seq_2^w), 5 (seq_3^w) and 7 (seq_4^w) steps. To estimate the time that one sequence of activities takes, we utilize a time series t_i (i=1, 2, 3,...) to denote the arrival time of the ith request in the activity sequences. The time series a_i is defined as t_{i+1}-t_i to denote the inter-request time of the ith and i+1th requests in a session. Since the ith request and i+1th request is envisioned as two states, the inner-request can be also called as the transition time. For two specific requests Req_1 and Req_2 in a certain sequence, let $\bar{a}$ denote the mean transition time from Req_1 to Req_2. For instance, in sequence seq_3^w, the mean transition time $\bar{a}$ from *BankInnerTransferConfirm* to *BankInnerTransfer* can be calculated as $\bar{a} = t_5 - t_4 = 40.3$s. Then the mean transition time of a sequence can be calculated as the sum of all mean transition times of two neighboring states in the sequence. By calculation, seq_1^w, seq_2^w, seq_3^w and seq_4^w take 103.4s, 54.1s, 81.3s and 181.9s.

Table 1: Pearson correlation coefficient of session length and number of transactions (T: Transaction Types, N: Number of sessions, R: Correlation Coefficient)

T	N	R	T	N	R
1	364	0.80	7	7	1.0
2	94	1.0	8	7	1.0
3	8232	0.91	9	5	0.98
4	302	0.98	10	47	0.99
5	39	0.97	11	26	0.96
6	60	1.0			

In our analysis, we find that the session length of transaction sessions with the same dominating transaction type is heavily correlated with the number of transactions as shown in Figure 2. It can be interpreted that a long session contains multiple cycles of one type of payment operation.

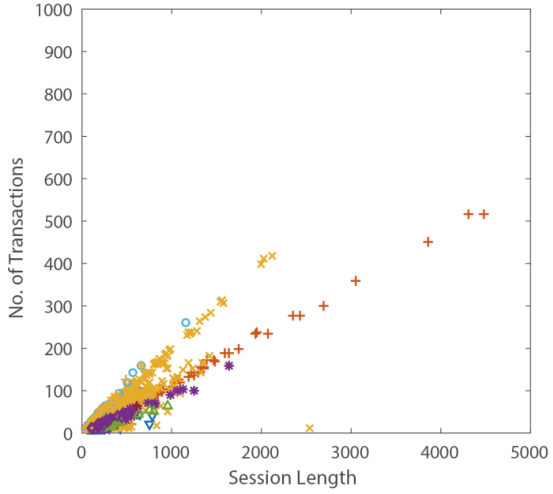


Fig. 2: Relation of session length and the number of transactions.

We use Pearson correlation coefficient to measure the significance of the correlation. The correlation coefficient (R) ranges from -1 to 1. A value of 1 implies that a linear equation describes the relationship between the session length and the number of transactions perfectly. Their Pearson correlation coefficients of all the 11 transaction types are shown in Table 1. When users repeat certain transaction tasks, they will periodically take the same steps.

3.2 Similarity measurement

To detect abnormal behavior, it is important to choose a measurement method to quantify the similarity among clickstreams, i.e., the distance among clickstreams. Therefore, we discuss methods to computing the distance between activity sequences in this section.

First, we formalize a user clickstream sequence as $S = (s_1\ s_2 \ldots\ s_i \ldots\ s_n)$, where s_i is the i^{th} element in the sequence and define T_k as the set of all k-grams(k consecutive elements) in S: $T_k(S)=\{\text{k-grams}|\text{k-grams}=(s_i s_{i+1} s_{i+k-1}), i \in [1, n + 1 - k]\}$. For sequences S_1, S_2 and a chosen k, we compute the set of all possible subsequences from both sequences as $T = T_k(S_1) \cup T_k(S_2)$. Next, we count the frequency of each subsequence within each sequence i(i=1,2) as array $[c_{i1}, c_{i2}, \ldots\ c_{in}]$ where n = |T|. Finally, we can get the similarity based on the distance of between S_1 and S_2.

Euclidean Distance: The distance can be computed as the normalized Euclidean distance

$$D_{eu}(S_1, S_2) = \sqrt{\sum_{j=1}^{n} (c_{1j} - c_{2j})^2}$$

Then, the similarity of the two sequences

$$C_{eu}(S_1, S_2) = 1 - sigmoid(D_{eu}(S_1, S_2))$$

Cosine Similarity: The similarity can be computed as the cosine similarity between the two arrays.

$$C_{cos}(S_1, S_2) = \frac{\sum_{i=1}^{n} c_{1i} \times c_{2i}}{\sqrt{\sum_{i=1}^{n} (c_{1i})^2} \times \sqrt{\sum_{i=1}^{n} (c_{2i})^2}}$$

Mixed Similarity: From our analysis, we find that the transaction numbers is heavily correlated with session length(n), which indicates that the cosine similarity is a good method to characterize different types of transaction sessions. However, the number of one type of transactions in a session also makes sense to judge the abnormality, e.g., in one session, 100 transfers have the different abnormality than only one. Therefore, we promote a method to mix these two measurements together.

$$C_{mix}(S_1, S_2) = C_{cos}(S_1, S_2) \times sigmoid(D_{eu}(S_1, S_2))$$

3.3 Anomaly Detection

To exploit the small amount of abnormal samples, we develop an unsupervised abnormal detection system. The key idea is to build a clustered sequence similarity graph. Each user's clickstream is represented by a single node. Every pair of nodes is connected by a weighted edge, where the weight is the similarity distance between the sequences. In Section 3.2, we have defined models of clickstreams as well as metrics for computing the distance between them.

After building a sequence similarity graph, we partition the graph via Louvain Method. Louvain Method is an efficient method to extract communities from large networks created by Blondel[3]. The method is a heuristic method that is based on modularity optimization.

At last, we color all clusters that include a seed sequence as "abnormal", while uncolored clusters are assumed to be normal. The whole progress is summarized as Algorithm 1.

Algorithm 1 Detection Algorithm

1: initialize seeds,vertexes,edges; // computing the weight of edges using similarity measurements;
2: G ⇐ buildGraph(edges, vertexes);
3: abnormals ⇐ ∅;
4: clusters ⇐ louvainCluster(G); // dividing the graph by Louvain method;

5: for each clusteri ∈ clusters do
6: vers ⇐ clusteri.getVertexes();
7: if vers ∩ seeds ∅ then
8: abnormals ⇐ abnormals ∪ vers;
9: end if
10: end for

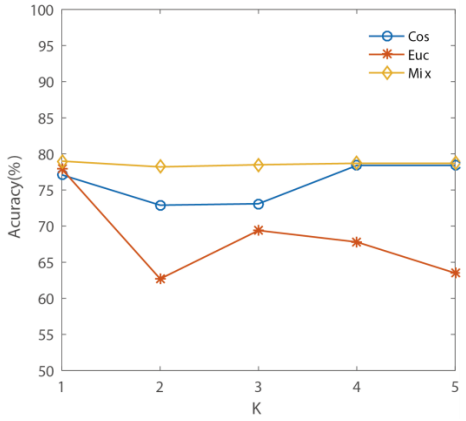
4. Experiments Analysis

To evaluate the effectiveness of the similarity measurement we proposed, we design two groups of experiments as in Table 2. Because of the restraint of time to manually verify the analysis results, all sessions in group A have session length more than 500. In group B, we randomly select 500 sessions with session length between 100 and 500 and add them to group A.

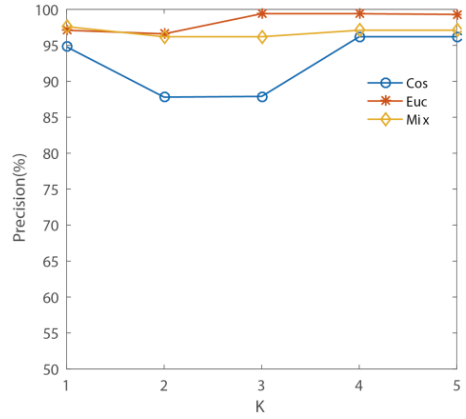
Table 2: Dataset description

Group	No. of sessions	No. of abnormal sessions
A	376	276
B	876	294

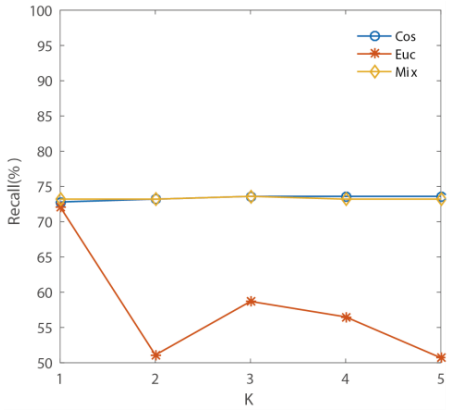
In our analysis, a True Positive (TP) is an abnormal transaction correctly labeled as abnormal. False Positive (FP) is a legitimate transaction wrongly labeled as abnormal, a False Negative (FN) is an abnormal transaction wrongly labeled as legitimate, and a True Negative (TN) is a legitimate transaction correctly labeled as normal.



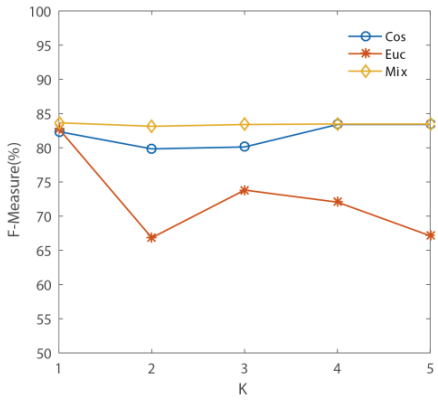
(a)



(b)



(c)



(d)

Fig. 3: Performance of different measurement methods with different k in Group A.

We start with a basic evaluation of system performance on data of group A. As shown in Figure 3, the performances (accuracy, precision, recall, F-measure) of these approaches vary for Euclidean distance method and cosine similarity method, when the k varies. However, the mixed similarity method relatively keeps the stability and has a high performances comparing to that of two methods above. And it gets a highest score (83.7%) on F-measure when k=1, which shows that increasing k offers no improvement in the model but introduces extra computation overhead.

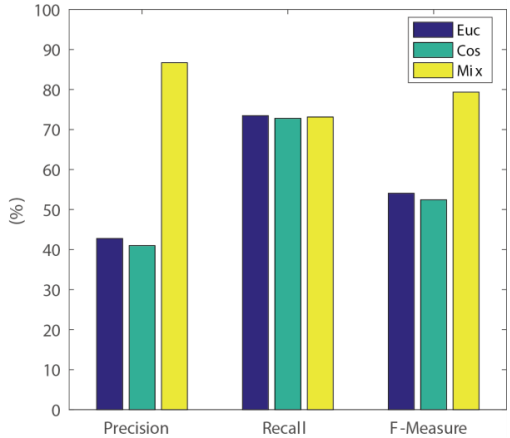


Fig. 4: Performance of different measurement methods with k=1 in Group B.

Next, we examine the system performance of these detection methods when 500 sessions (session length between 100 and 500) are added to the sequence similarity graph. From Table 4, we can find that these newly added sessions have more normal sessions. In Figure 4, we show the results of the detection methods with k=1. The precision scores of Euclidean distance method and cosine similarity method have dropped significantly compared to the results in group A. Finally, only the mixed similarity method have a relatively good F-measure score (79.3%), which indicates that the mixed similarity method we proposed has a better

performance than only Euclidean distance or cosine similarity method, especially when the normal data samples more than abnormal samples.

5. Related Work

There are few published works about abnormal detection within the domain of online banking, due to the privacy, secrecy and commercial interest. Usually, these works detected the online banking fraud based on the model of user history behavior [4, 5, 6, 7, 8]. Their research models the behavior of each customer and monitors whether it deviates from normal behavior [9].

Wei *et al.* introduced a systematic online banking fraud detection method using transaction data from a large Australian bank [6], but they did not provide any detailed analysis for the online banking customer behavior. Carminati *et al.* [8] developed a semi-supervised and unsupervised fraud and anomaly detection method based on a real-world dataset of a large Italian national bank. His system design is guided by behavior analysis, but his work only describes the distribution of the amount and transaction frequency.

Compared to these studies, our research relies on the dataset that includes more details about the user behavior during the transactions, e.g., user click stream. Therefore, we can promote the appropriate method to compute the similarity of user behaviors. Moreover, because of the lack of publicly available and real-world frauds, most these works resort to synthetically generated frauds. Our work based on ground truth data reveals some of the abnormal behavior that is happening.

6. Conclusion

In this paper, we have proposed an approach to catch the bad behaviors similar to the seeds in online banking transactions, by dividing the similarity graph of user click sequences. Our work is based on analyzing the click behavior of online banking users on personal transaction data, which is collected from a large bank in China.

Our contribution in this paper is to analyze the user behavior and promote a mixed method to measure the similarity between user behaviors. And we show that our method can improve the performance on the underground dataset, comparing to Euclidean distance and cosine similarity.

In conclusion, our work will be helpful to improve the abnormal detection of the online banking.

7. References

[1] Carminati, Michele, et al. Security Evaluation of a Banking Fraud Analysis System. ACM Transactions on Privacy and Security (TOPS) 21.3 (2018): 11. DOI= https://doi.org/10.1145/3178370

[2] Benevenuto, Fabrłcio, et al. Characterizing user behavior in online social networks. Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement. ACM, 2009. DOI= http://doi.acm.org/10.1145/1644893.1644900

[3] Blondel, Vincent D., et al. Fast unfolding of communities in large networks. Journal of statistical mechanics: theory and experiment 2008.10 (2008): P10008. DOI= http://dx.doi.org/10.1088/1742-5468/2008/10/P10008

[4] Karlsen, Kare Nordvik, and Tarje Killingberg. Profile based intrusion detection for Internet banking systems. MS thesis. Institutt for datateknikk og informasjonsvitenskap, 2008.

[5] Kovach, Stephan, and Wilson Vicente Ruggiero. Online banking fraud detection based on local and global behavior. Proc. of the Fifth International Conference on Digital Society, Guadeloupe, France. 2011.

[6] Wei, Wei, et al. Effective detection of sophisticated online banking fraud on extremely imbalanced data. World Wide Web 16.4 (2013): 449-475. DOI= http://dx.doi.org/10.1007/s11280-012-0178-0

[7] Cabanes, Guenael, Younes Bennani, and Nistor Grozavu. Unsupervised learning for analyzing the dynamic behavior of online banking fraud. 2013 IEEE 13th International Conference on Data Mining Workshops (ICDMW). IEEE, 2013. DOI= http://doi.ieeecomputersociety.org/10.1109/ICDMW.2013.109

[8] Carminati, Michele, et al. BankSealer: A decision support system for online banking fraud analysis and investigation. computers & security 53 (2015): 175-186. DOI= https://doi.org/10.1016/j.cose.2015.04.002

[9] Jyothsna, V. V. R. P. V., VV Rama Prasad, and K. Munivara Prasad. A review of anomaly based intrusion detection systems. International Journal of Computer Applications 28.7 (2011): 26-35.