Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 4434714, 14 pages
https://doi.org/10.1155/2022/4434714

Research Article

A Detection Method for Abnormal Transactions in E-Commerce Based on Extended Data Flow Conformance Checking

Yadi Wang[1,2] **Wangyang Yu**[1,2] **Peng Teng**[3] **Guanjun Liu**[4] **and Dongming Xiang**[5]

[1]*Key Laboratory of Modern Teaching Technology, Ministry of Education, Xi'an 710062, China*
[2]*School of Computer Science, Shaanxi Normal University, Xi'an 710119, China*
[3]*School of Journalism and Communication, Shaanxi Normal University, Xi'an 710119, China*
[4]*The Department of Computer Science and Technology, Tongji University, Shanghai 201804, China*
[5]*The School of Information Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China*

Correspondence should be addressed to Wangyang Yu; ywy191@snnu.edu.cn and Peng Teng; tengpeng@snnu.edu.cn

Received 24 September 2021; Revised 13 October 2021; Accepted 15 October 2021; Published 4 January 2022

Academic Editor: Deepak Kumar Jain

With the development of smart devices and mobile communication technologies, e-commerce has spread over all aspects of life. Abnormal transaction detection is important in e-commerce since abnormal transactions can result in large losses. Additionally, integrating data flow and control flow is important in the research of process modeling and data analysis since it plays an important role in the correctness and security of business processes. This paper proposes a novel method of detecting abnormal transactions via an integration model of data and control flows. Our model, called Extended Data Petri net (DPNE), integrates the data interaction and behavior of the whole process from the user logging into the e-commerce platform to the end of the payment, which also covers the mobile transaction process. We analyse the structure of the model, design the anomaly detection algorithm of relevant data, and illustrate the rationality and effectiveness of the whole system model. Through a case study, it is proved that each part of the system can respond well, and the system can judge each activity of every mobile transaction. Finally, the anomaly detection results are obtained by some comprehensive analysis.

1. Introduction

E-commerce has developed rapidly and is now in a golden age of digital economy. E-commerce has been used by every individual or company conforming to the times to sell or buy goods and services in the form of electronic payment [1]. According to statistics, the amount of mobile transactions is increasing significantly year by year [2]. However, anomaly events and attacks always occur in the transaction system, especially in mobile applications [3, 4], and it is difficult for an e-commerce system to deal with fraud [5]. That is because a component is often shared by multiple software, and many threats and attacks are related to them, such as sniffing, spoofing, and malware [6]. Moreover, if malicious attackers take advantage of the knowledge gap of multiple participants, it is difficult to ensure the security of the payment processes [7, 8]. Therefore, how to ensure a healthy and security trading environment is always a key consideration for e-commerce platforms. What is more, the development of a comprehensive fraud detection system is of great importance to many organizations and companies [1, 9]. Many fraud detection methods of electronic transactions have been proposed based on that.

Abnormal transactions or frauds mainly refer to the use of some means to defraud money or commodities [7]. The task of detecting abnormal transactions by machine learning can be seen as a binary classification problem, and this kind of method relies on historical data. The issues such as skew distribution and concept drift present challenges for machine learning technology to extract meaningful patterns from the historical data [5]. Some complex machine learning methods such as support vector machines, neural networks, and deep learning are black box that means it is difficult for them to interpret fraud patterns, while interpretability is the key to designing fraud prevention mechanisms [9]. Zheng et al. proposed an improved machine learning method of

TrAdaBoost (ITrAdaBoost), which is more suitable to deal with the concept drift problem under different data distributions [10]. It is important and necessary to use technology that helps detect anomalies and prevent fraud. Fraud prevention includes taking measures to prevent fraud from occurring, or respond quickly to fraud and prevent losses in time [2]. However, real-time fraud prevention is not easy [2]. Data mining techniques make it possible to find valuable patterns in data sets. However, some data mining methods are not process centric [11].

Electronic commerce systems are complex, highly concurrent, and distributed. The formal analysis method is an efficient knowledge representation and discovery tool, such as formal concept analysis (FCA) [12]. Data errors and state inconsistencies are often inseparable from the abnormal events in an online shopping system [13]. Petri net and its extensions are useful tools for modeling and simulating business processes, which can describe and reflect the trading process of multiparties dynamically [14, 15]. However, the general Petri nets cannot exactly express the data operations of concurrent read, coverable write, and some other operations well. That is why a Petri net with data operations (PN-DO) is promoted [16]. PN-DO can analyse data operations in a concurrent system and check their data inconsistency, missing data, and other data flow errors. Due to the fact that a transaction fraud detection system is inseparable from data-driven methods [9], it requires data analysis methods to obtain data attributes. Data Petri net (DPN) is a kind of net that adds some data attributes to a Petri net, and it has been widely applied to different business process optimizations [17]. Because the data flow of the e-commerce model often has a large number of concurrent read and write operations, we should extend DPN to model data operations besides data attributes. The existing data analysis methods are not enough to cope with the abnormal data interaction among the activities of the e-commerce transaction model based on the process perspective and the fusion detection of user behavior habits, and thus, we hope that our extension can also handle these problems.

How to analyse and predict the online transaction process in real-time is the focus of our concern. In this paper, the users with abnormal behavior combinations in the overall process of electronic transactions are regarded as abnormal users. We take e-commerce fraud detection as a background for modeling and case analysis. This paper mainly researches on the data interaction of e-commerce transactions and analyses its anomalies. An e-commerce system is a distributed system and a multiparty data interaction process. We choose Petri net as our analysis method because it is an effective modeling tool and can dynamically reflect data interactions between various activities and resources [13, 18]. Based on that, this paper combines PN-DO with DPN and proposes an extended net, i.e., Extended Data Petri net (DPNE). Because it can combine the functions of algorithms, the function of data flow analysis can be expanded, so that activities can be expressed and analysed more intuitively. This approach ties the control flow more closely with the data flow. Our research mainly focuses on the data flow analysis technology of DPNE. It is mainly used to solve the integrated data flow processing of transactions. We propose two algorithms to describe how it works, i.e., outlier extraction and abnormal order detection.

We propose an abnormal orders detection algorithm which refers to conformance checking [17] and a full sequence comparison algorithm [19] to realize dynamic full tracking of the transaction process. This mechanism can dynamically reflect the data interaction and the state of each activity from the perspective of control flow and data flow. Finally, we combine the standard model predefined by experts to analyse the comprehensive situation of all activities so as to analyse and judge whether the current transaction is abnormal or not.

The main contributions of this research are as follows:

(1) PN-DO [16] and DPN [17] are employed to the formalization of e-commerce transaction systems, and their whole business process is modeled and analysed by an Extended Data Petri net (DPNE);

(2) By analyzing the model structure of DPNE, the whole system is optimized. Then, some abnormal detection algorithms for integrated data flow processing of orders are proposed. Furthermore, the running states of this system are analysed by the algorithm in [20].

The paper is organized as follows. Section 2 presents an overview of the current researches status on anomaly detections and other correlation analysis methods of e-commerce. Section 3 introduces the related basic concepts and proposes DPNE. In Section 4, a pattern matching method is integrated into our conformance checking, and the working principle is described by some algorithms. Section 5 presents the method of outlier extraction and analysis proposed in the previous section through a case study in detail. Finally, the paper is summarized in Section 6.

2. Related Work

At present, there are many kinds of researches concerning anomaly detection of the e-commerce system [21–25]. There are two key points in the anomaly detection technology. One is how to establish a library of normal behavior patterns based on historical information, and the other is how to compare the current behavior patterns with normal behavior patterns [21]. Anomaly detection often uses data mining technologies, the purpose of which is to extract unknown and valuable patterns or rules from a large amount of data. The most widely used data mining algorithms include data classification, correlation analysis, and sequence mining [9, 10]. In order to cope with the diversification of users' transaction behaviors, Zheng et al. proposed the logic graph BP (LGBP), which is a total-order-based model. It is used to represent the logical relationship of transaction record attributes [22]. With machine learning classification algorithms, the credit scoring model is used to predicting the customer default in e-commerce by analyzing the historic data [25]. However, these related studies always focus on historical

data, lacking analysis of concurrent operations. E-commerce systems are full of concurrent activities. Therefore, a comprehensive method that is equipped with data analysis and real-time monitoring and investigation of each activity of control flow is still needed.

There are some studies on process modeling and analysis, e.g., Petri nets are used to build e-commerce business processes [13]. However, most of these researches are to simulate the business process of e-commerce transactions. Then, they analyse the rationality of their models so as to ensure the transaction properties of the e-commerce business process, rather than use some formal modeling tools to assist outlier analysis and decision-making [26–28]. It is essential to use formal methods to model and analyse the information exchange in a trading process of e-commerce transactions. For example, Bartosz et al. [29] used Colored Petri net to model cyber threats directed at computer systems formally and then solved up-to-date problems related to threats detection and prevention. Logical Petri net (LPN) is proved to be a great tool to simulate an e-commerce system [14]. PN-DO not only retains the benefits of the prototype Petri net but also is suitable for modeling and analysis of large-scale concurrent reads and writes [16]. Related cases have proved that it is suitable to model and analyse some business processes effectively, e.g., publications and multithreads [30].

Abnormal detection is also related to conformance checking. The basic idea of process mining is to diagnose business processes by mining event logs to obtain knowledge. As one of its applications, conformance checking is widely used to investigate and quantify the discrepancies between the real execution logs and process models, so as to quantify the difference between the real behaviors and the behaviors determined by models [31]. To reveal this kind of biases, data attributes are necessary [32]. Existing conformance checking methods include conformance checking that focuses on control flows [31, 33] and techniques for diagnosing data-related deviations [34–37]. However, these methods cannot be directly applied in the field of e-commerce and do not detect anomalies in multiple activities in a transaction process, such as user behavior detection or a comprehensive decision that takes into account multiple transitions [32, 34, 36, 37]. Conformance checking can analyse data flows to explain whether current data is suitable for some specific transitions or not and discover the related decision points [17]. In order to improve business processes, van der Alst et al. compared the real behaviors of an information system (or its users) with the expected behavior by delta analysis, e.g., certain activities were performed by specific users [34]. Moreover, data plays an important role. For example, routing in a process is usually determined by data, which indicates that the control flow is to some extent data dependent [32]. de Leoni et al. [32] proposed a technique to detect the conformance of data-aware process models. They used Data Petri nets (DPN) to model data variables, guards, read and write operations, and matched data attributes of a single activity. DPN incorporates various data attributes and further expands the function of conformance checking. When multiple activities with data attributes compete for the unique resource, this is a decision point, and the condition of an activity triggered by the decision point is a rule [36, 37]. Haarmann et al. [35] used Colored Petri nets to describe a decision-aware process formally and then utilized temporal logics to check compliance rules. Process mining focuses on the overall process of a transaction, and conformance checking of data flows also greatly improves the ability compared to control flows individually. However, it is not enough to directly detect and analyse the abnormal behavior in the process of e-commerce transactions.

The above studies mainly focus on the modeling and analysis of resources, data mining of historical transaction data, or matching of business rules for a single activity. Compared with them, our method is a process-based, real-time, and dynamic data analysis method. It can predict each order in time and intercept the abnormal ones. Moreover, normal process-based trading patterns of users are recorded and regarded as one of the next rules, resulting in a more personalized matching criteria for each user. Furthermore, the reference criteria of this method will intelligently improve as users' patterns change.

Based on the e-commerce transaction interaction data, as well as various abnormal events that may exist in reality, this paper adopts formalization and data analysis methods to conduct real-time analysis and research on abnormal behaviors in the transaction process. In this study, DPNE is used to formalize an e-commerce trading and anomaly detection system. The interaction information of multiple terminals and system servers is used to detect algorithm fusion for a single activity and analyse the combination of multiple activities to complete the capture and comprehensive analysis of outliers in the whole process.

3. Preliminary

This paper mainly focuses on the whole process from activities of buyers' logging in to the transaction done. The system runs dynamically. By modeling and analysing the interactive data in this system, the current transaction can be monitored in real time. Therefore, this system can monitor and catch suspicious anomalies during the trading process and make a timely and comprehensive judgment on the current order at the end of the transactions and intercept abnormal orders and users.

Figure 1 shows the data interaction of an e-commerce trading system, which has a multiparty participation process [38]. It consists of several different types of resources, e.g., buyers, sellers, e-commerce trading platforms, cashier servers, and third-party servers. The data of interactions between these five parties are recorded in the data center. In this section, we will introduce the Extended Data Petri net, which is used to model e-commerce transaction processes, and some relevant works can be seen in [7, 8, 38].

3.1. Modeling Approach. Petri net is widely used as the basis for the formal modeling of systems. In reality, it is usually extended to describe more complex processes, e.g., Time Petri net [39] and logical Petri net [40]. In order to effectively simulate an e-commerce transaction process, we

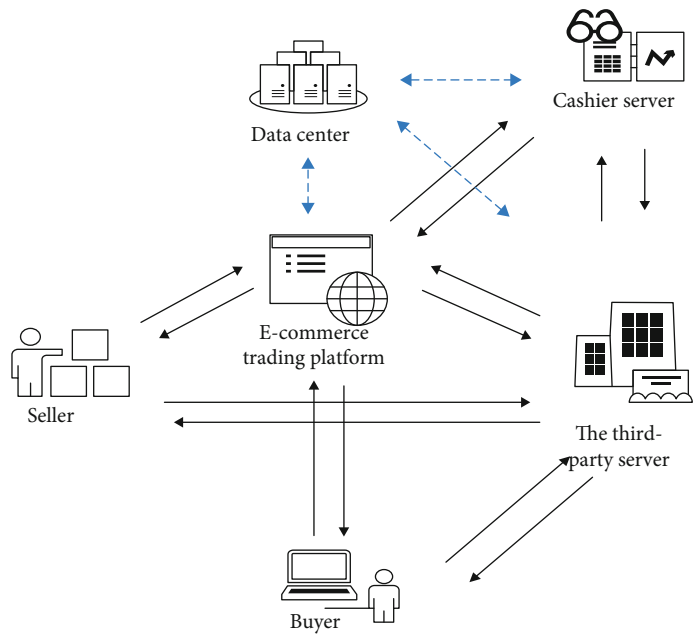

Figure 1: The resource interaction process of e-commerce.

propose Extended Data Petri net, which is based on Petri net with data operations (PN-DO) [16] and Data Petri net (DPN) [17].

Definition 1. (PN-DO). A PN-DO is a 4-tuple net and PN – DO = $(P_c \cup P_v, T, F_c \cup F_r \cup F_w \cup F_d, M_0)$ [16], where:

(1) P_c is a set of control places, and P_v is the set of data places, $P_c \cap P_v = \varnothing$

(2) T is a set of transitions;

(3) $F_c \subseteq P_c \times T \cup T \times P_c$ is a set of control arcs;

(4) $F_r \subseteq P_v \times T$ is a set of read arcs;

(5) $F_w \subseteq T \times P_v$ is a set of write arcs;

(6) $F_d \subseteq P_v \times T$ is a set of delete arcs and $F_r \cap F_d = \varnothing$

PN-DO can guarantee not only the correctness of control flows but also the correctness of data flows [16]. Thus, it is more suitable to describe a process model with data interactions than traditional Petri nets.

Definition 2. (DPN). A DPN is a 8-tuple net and DPN = (P, T, F, V, U, R, W, G) [17], where:

(1) (P, T, F) is a Petri net that contains a set of places, transitions, and arcs;

(2) V is a set of data variables;

(3) R is a read function, and it labels each transition with a set of variables that need to be read, which is represented as $R \in T \longrightarrow \rho(V)$

(4) W is a write function, and it labels each transition with a set of variables that need to be written, which is represented as $W \in T \longrightarrow \rho(V)$

(5) G is a guard function that defines canonical variable rules for each transition, which is represented as $G \in T \longrightarrow$ Formulas $(V_W \cup V_R)$

PN-DO is essentially a control-flow-oriented model. To further describe and analyse the data flows in the business process of e-commerce trading, data attributes are needed. DPN is a Petri net that introduces data attributes. It can analyse data in each activity and interpret why individual cases take a particular way [17]. In order to get a better data analysis, DPNE is defined by combining the above advantages of nets.

Definition 3. (DPNE). A DPNE is a 7-tuple $N = (\text{PN} - \text{DO}, V, U, R, W, G, \Phi)$, in which:

(1) *P*N-DO is a Petri net with data operations, which is $(P_c \cup P_v, T, F_c \cup F_r \cup F_w \cup F_d, M_0)$;

(2) V is a set of data variables that are used in the transitions; the name and number of V are consistent with that of P_v, i.e., the variables defined in PN-DO are consistent with the data variables in V

(3) U is a function that defines the range of each value, i.e., D_v is the domain of a variable value v, and the value of all variables should be within the range defined, i.e., for each value $v \in V$, it satisfies $U(v) = D_v$

(4) R is a read function $R \in T \longrightarrow \rho(V)$, which indicates the sets of defined variables that need to be read for each transition;

(5) W is a write function $W \in T \longrightarrow \rho(V)$, which indicates the sets of variables that need to write for each transition;

(6) G is a guard function $G \in T \longrightarrow \text{Rules}(V_W \cup V_R)$. The guard is represented by some combination rules of reading variables and writing variables;

(7) Φ is an algorithm or function, which processes a set of reading or writing variables of the transition with specified algorithm functionality, i.e., $V(t) \longrightarrow^{\phi(t)} V_{\text{new}}(t)$

We use quadruple (t, r, w, φ) to describe each transition and the situation of reading and writing variables in a DPNE. Notation (t, r, w, φ) means transition t have read function r, write function w, and algorithmic functional parameter φ. In a DPNE, we just introduce data attributes to the PN-DO model without changing the nature of its control flows. The detection of reachability and data flow errors is consistent as described in [30].

Definition 4. (Valid bindings). If a quadruple (t, r, w, φ) meets the following conditions, then it is considered as valid:

(1) $t \in T$ meets control enabledness [30] at a marking M, i.e., $M[ct>$, which means each of the input control places$\cdot t$ contains tokens, and then, t is enabled at the markings M with control places, denoted by $\forall t \in T$, if $\forall p_c \in \cdot t$: $M(p_c) \geq 1$

(2) $t \in T$ meets data enabledness [30] at a marking M, i.e., $M[dt>$, which means each of the input data places$\cdot t$ contains tokens, and then, t could be enabled under the markings M with data places, denoted by $\forall t \in T$, if $\forall p_d \in \cdot t$: $M(p_d) \geq 1$

(3) $r \in R(t) \longrightarrow U_v$ and $w \in W(t) \longrightarrow U_v$

(4) $\forall v \in R(t)$: $r(v) \in U_v$

(5) $\forall v \in W(t)$: $w(v) \in U_v$

(6) $\forall v \in \Phi(t)$: $\varphi(v) \in U_v$, i.e., the input variables of the function are predefined;

(7) $G(t)$ is evaluated as true with the input and output variables v of t

There are some basic notations [30] as follows:

(1) ${}^c t$ represents a set of control places whose postset is transition t

(2) t^c represents a set of control places whose preset is transition t

(3) ${}^r t$ represents a set of data places that are read by transition t

(4) t^w represents a set of data places that are written by transition t

(5) ${}^d t$ represents a set of data places that are deleted by transition t

Control place reflects control flow, and data place reflects data flow. A binding that satisfies the above conditions is valid. That means a transition t with a valid binding $B = (t, r, w, \varphi)$ can be fired, and the firing rules of DPNE are described as follows: there are two types of places in DPNE, one is the control places and the other is the data places. For transitions whose preset is control places, the enabled transitions will consume each of its token from the preset and generate one token to the postset. That is, if $\forall t \in T$, $\forall p_c \in \cdot t : M(p_c) \geq 1$, then $M_{\text{new}}(p_c) = (M(p_c) \setminus \cdot t) \cup t \cdot$.

For transitions whose preset is data places, there are three situations. If a transition t with write-set t^w, then t will generate tokens to each data place of the write-set; if a transition with delete-set ${}^d t$, then it will consume tokens from the data places of the delete-set, while, if a transition with read-set ${}^r t$, it means that the token in the read-set ${}^r t$ will not be added or subtracted. And the variables related to the places can be read multiple times, and the arcs are represented by dotted lines.

These can be written formally [30] as follows:

$$M'(s) = \begin{cases} M(s) - 1, & \text{if } s \in {}^d t, \\ M(s) + 1, & \text{if } s \in t^w \wedge M(s) = 0, \\ M(s), & \text{otherwise.} \end{cases} \quad (1)$$

$(\text{DPNE}, (M, s))[b > (M', s')$ describes a valid binding b at a marking (M, s) may occur. A new marking (M', s') is generated after firing this transition.

In this paper, the business processes of e-commerce transaction systems are modeled and analysed by DPNE, and the detection function of users' static and dynamic behaviors is also integrated into this process.

3.2. The DPNE Model of E-Commerce Transaction. In this subsection, we present a case of electronic trading systems and their related outlier extraction and further propose detection algorithms to illustrate how the system modeled by DPNE can catch outliers based on real-time input data.

Figure 2 shows the DPNE model of an e-commerce transaction process. This model introduces the e-commerce transaction process from the time when a buyer logs in to the client to the end of this transaction. Table 1 defines the meaning of each variable attributes. Table 2 describes the data interactions of the Petri net in Figure 1. During the whole process, the model collects and addresses data in real time. On the e-commerce trading platform, multiple buyers and sellers interact concurrently, and several servers access simultaneously. Due to the fact that we only analyse the working principle of e-commerce systems, we adopt a single-user mode.

Figure 2 concentrates on an interactive process of real-time data inputs and outputs for a buyer from the start of the logging client to the end of the order. It is modeled by DPNE. The transaction platform requires the buyer to have sufficient personal information. The buyer's browsing, viewing, adding to the shopping cart, and other operations are

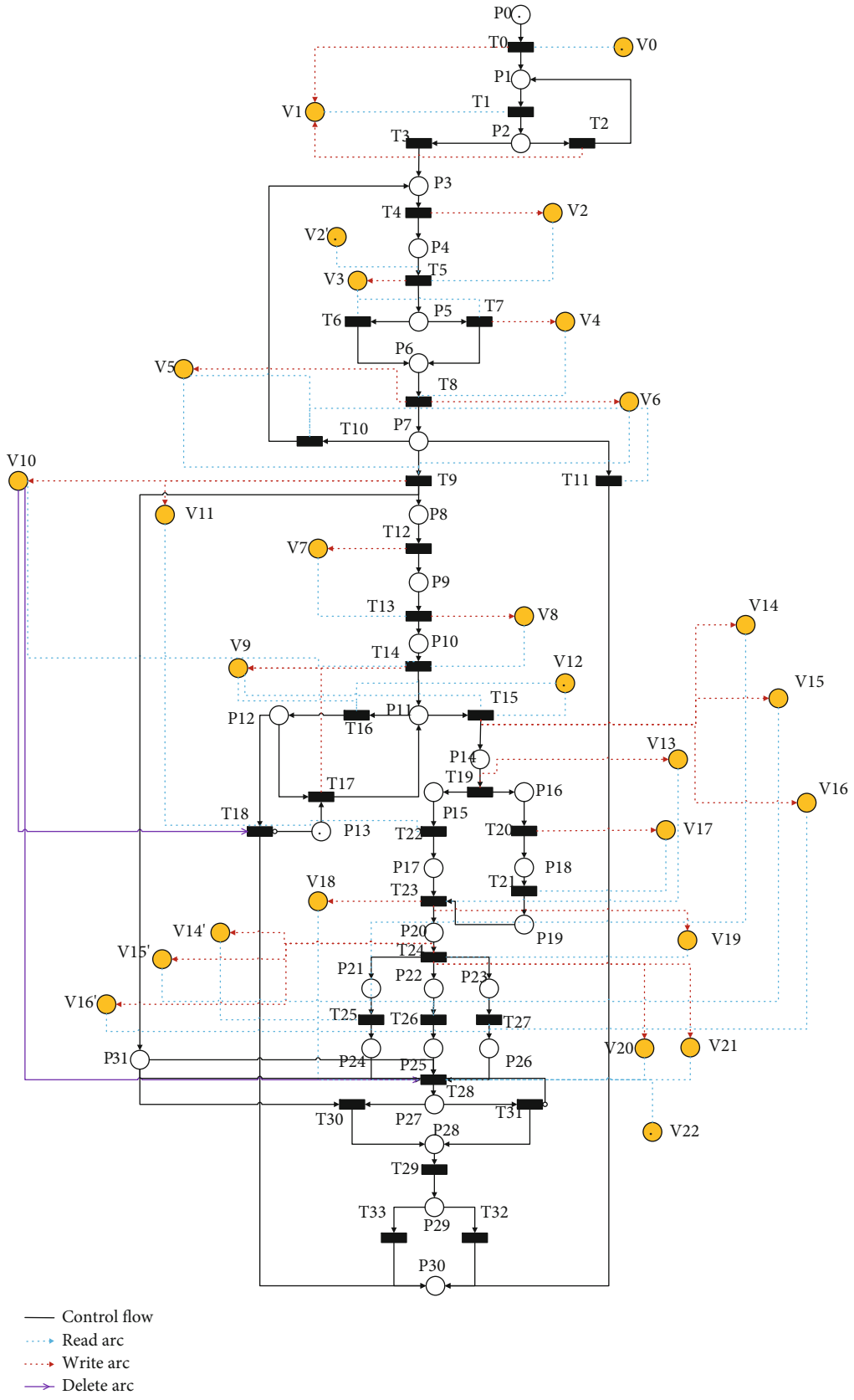

Figure 2: DPNE model of e-commerce system.

Table 1: Variables and meanings.

Variables	Meaning
V_0	Buyer_ID
V_1	BuyerInfo
V_2	Current dynamic behavior
$V_2{}'$	Normal dynamic behavior pattern
V_3	Dynamic behavior matching results
V_4	GoodsInfo
V_5	Intention to sell
V_6	Intention to buy
V_7	TN(CS_signature_status)
V_8	TN(S_signature_status)
V_9	Real-time password input
V_{10}	Payment_status_unpaid
V_{11}	Seller_ID
V_{12}	Password
V_{13}	Payment completed CS signature
V_{14}	Total_amount
V_{15}	Address
V_{16}	Goods_Type
$V_{14}{}'$	Total_amount_paid
$V_{15}{}'$	Address_paid
$V_{16}{}'$	Goods_Type_paid
V_{17}	Notice of payment completion
V_{18}	Total duration of transaction
V_{19}	Payment_status_paid
V_{20}	LoginIP
V_{21}	LoginDT
V_{22}	Normal static attribute pattern

recorded in the client terminal. The current operational behavior is analysed, and a warning is sent to the users who deviate significantly from the normal dynamic mode. After the order is submitted, the cashier server establishes a unique transaction certificate for it. The payment password is entered a maximum of 2 times. Otherwise, the order will be automatically terminated. After the payment is done, the information of successful payment is sent to the third-party cashier and seller. Then, the third-party cashier sends the information to the buyer. After that, the status of the order is changed to paid, and then, the submitted order and paid order are rechecked. Meanwhile, the static behavior data is matched for the current real-time static behavior. At the same time, the full sequence comparison algorithm [19] is used to process the current real-time static behavior so as to get the matching result. Finally, the overall data interaction is summarized and analysed to determine whether the current transaction is abnormal or not.

Control flows focus on the partial orders of tasks, while data flows relate to data operations. Our model can reach terminated states (transactions completed or terminated for some reason). Data flow errors are defined in [30], including missing data and inconsistent data. Data flow errors can be associated with exceptions in the process. Through the verification of the structural properties, the rationality and correctness of business process are guaranteed. They provide a theoretical basis for the follow-up work.

4. Outlier Extraction and Detection Algorithm

In this section, an algorithm based on the DPNE model is proposed on the basis of the conformance checking algorithm [41]. It is used to judge whether the current transaction is abnormal. The DPNE model is simulated by the proposed algorithm and [20] is used to analyse the running state of the whole process.

4.1. User Behavior Matching. The behavior data of users recorded by the operations of the client terminals mainly contains the information of the users' behavior habits. These behavior data contain not only static attribute data (e.g., user ID and IP address) but also behavior data of dynamic operations. So as to make full use of data information, the characteristics of abnormal behaviors make it necessary to separate these two types of data for mining user behaviors. This is beneficial to the discovery of more subtle anomalies.

Pattern matching is an important research field for judging abnormal behaviors of users. A normal behavior pattern can be obtained by pattern mining algorithms, and then, the current behavior pattern is compared with the pattern in a normal behavior pattern library so as to achieve the purpose of detection [19]. By integrating the pattern matching algorithm into the conformance checking method of process mining, its functions are extended, and they are suitable for more application scenarios.

The user's behavior habits are hidden in the operation data, and each user's operation habits are different. Therefore, these operational data have a strong personality. When others use the same common devices, accounts, and IP addresses as real users to conduct e-commerce operations, they can make use of the behavioral patterns mined from the user's historical behavior data for pattern matching. Thus, they can find the differences between their current behaviors and users' habits.

As shown in Table 3, the user behavior data used in this method mainly falls into the following categories. Since it has a limited number of categories, the related user behavior can be marked and classified as integers.

4.2. User Behavior Analysis. Our user data is divided into static attribute data and dynamic behavior data. Thus, the detections for static attribute anomaly and dynamic behavior anomaly are carried out, respectively. After then, the existence of anomalies is judged.

Table 2: Graphic symbol in DPNE.

Transition	Meanings	Input data	Output data	Delete data	Φ
T_0	Log in	V_0	V_1	None	None
T_1	Check for completeness	V_1	None	None	None
T_2	More info	None	V_1	None	None
T_3	Completeness checking pass	None	None	None	None
T_4	User operation	None	V_2	None	None
T_5	Operation detection	V_2, V_2'	V_3	None	Recursive correlation
T_6	Warning	V_3	None	None	None
T_7	Submit orders	V_3	V_4	None	None
T_8	Order addressed	V_4	V_5, V_6	None	None
T_9	Order accepted	V_5, V_6	V_{10}, V_{11}	None	None
T_{10}	Invalid orders	V_5, V_6	None	None	None
T_{11}	Order canceled	V_6	None	None	None
T_{12}	TN created by CS	None	V_7	None	None
T_{13}	TN accepted by seller	V_7	V_8	None	None
T_{14}	Request payment and enter password	V_8, V_{10}	V_9	None	None
T_{15}	Password correct	V_9, V_{12}	V_{14}, V_{15}, V_{16}	None	None
T_{16}	Password wrong	V_9, V_{12}	None	None	None
T_{17}	Re-enter the password	None	V_9	None	None
T_{18}	Password error exceeds limit	None	None	V_{10}	None
T_{19}	Payment	None	V_{13}	None	None
T_{20}	Notify TP payment done	None	V_{17}	None	None
T_{21}	Notify buyer payment done	V_{17}	None	None	None
T_{22}	Notify seller payment done	V_{11}	None	None	None
T_{23}	Buyer click to view and update order status	V_{13}	V_{18}, V_{19}	None	None
T_{24}	Check and process payment orders	V_{19}	$V_{14}', V_{15}', V_{16}', V_{20}, V_{21}$	None	None
T_{25}	Check and process orders amount	V_{14}, V_{14}'	None	None	None
T_{26}	Check and process orders goods	V_{15}, V_{15}'	None	None	None
T_{27}	Check and process orders address	V_{16}, V_{16}'	None	None	None
T_{28}	Detect user static attribute	$V_{18}, V_{20}, V_{21}, V_{22}$	None	V_{10}	Full sequence comparison
T_{29}	Success purchase	None	None	None	None
T_{30}	Existing extra unpaid orders	None	None	None	None
T_{31}	Orders all paid	None	None	None	None
T_{32}	Abnormal transaction	None	None	None	None
T_{33}	Normal transaction	None	None	None	None

Table 3: User behavior categories and mathematical identifications.

Category	Search	Browse	Favorites	Add to shopping cart	View cart	View favorites	Submit orders
Identification	0	1	2	3	4	5	6

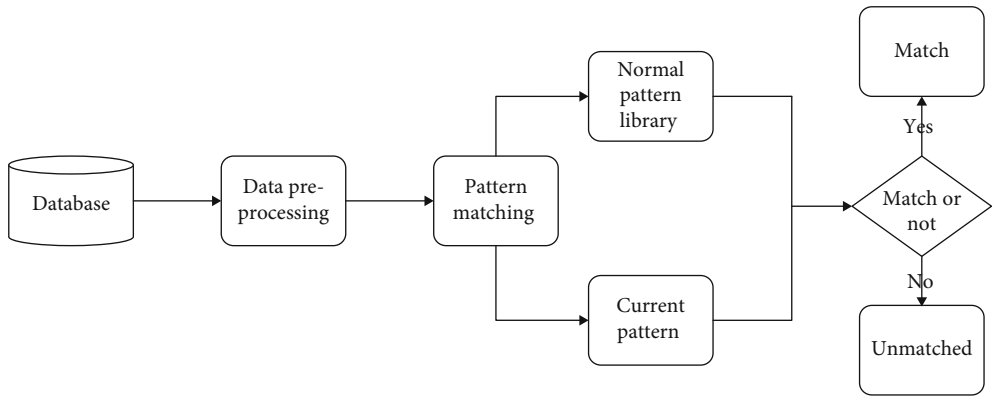

Figure 3: User data pattern mining and detection process.

Figure 3 shows the specific steps of anomaly detection for static attributes and dynamic behaviors of users [42], i.e.,

(1) Step 1: obtain normal historical data of users from user databases, preprocess the data, and transform it into mathematical symbols;

(2) Step 2: use data mining algorithms to obtain the user's normal data pattern library from the normal historical data and take it as the standard of normal states;

(3) Step 3: make a pattern comparison between the current user data and the user's normal data pattern library. If it is close to the normal pattern, the current user state is considered normal; otherwise, it is marked as an outlier.

With respect to the user static attributes, this paper chooses a full sequence comparison method [19] to compare the user's current static attributes with the static attribute pattern. As for the dynamic attributes, a recursive correlation algorithm [19] is used to compare patterns. Our method fully considers the factors of subsequences and calculates the similarity of two sequences more accurately.

4.3. An Anomaly Detection Algorithm Based on the DPNE Model. Although conformance checking and abnormal detection are different concepts, they have similar definitions of the concept of outliers. Conformance checking has the function of capturing abnormal data, which brings a new vision to the traditional method of abnormal detection. The identification of nonconforming traces is clearly valuable. Abnormal or deviant behaviors occur from time to time in the process of e-commerce transactions, i.e., the login IP changes when the buyer travels to buy goods, or the buyer uses someone else's account to pay. Moreover, there may be different judgment results in different scenarios for the same deviation. For example, a change in the transaction amount, it may be a fraud attack. It also may be a result of bargaining between a buyer and a seller, and the final payment amount is changed by the seller. Therefore, not all deviations are fraudulent.

Data-aware conformance checking is similar to the traditional consistency alignment operation in process mining [17]. Conformance checking matches each trace in the event log L to the process model DPN. The difference is that our purpose is not to achieve the optimal match of the minimum cost sum to preserve the event log and model information as much as possible. In this paper, our purpose is to use DPNE for anomaly detection of e-commerce transactions. While, any activity that does not conform to the model will be considered as an outlier, and a transaction with outliers is likely to be a fraud transaction. Moreover, to achieve a more complete anomaly detection function, it is essential to integrate the relevant similarity matching algorithm in this method.

Based on the DPNE model in the whole analysis process, we need to establish two algorithms. Algorithm 1 is proposed to analyse the matching of the data of each active data flow and its guards. Then, the outliers are extracted to obtain the current data flow pattern sequences. Algorithm 2 is to make a comprehensive analysis of the current transaction data flow sequence so as to judge whether the current transaction is normal or not. These two algorithms are described in detail below.

The first algorithm is used to do full sequence matching [19] for a single activity and to extract the matching sequence made up of multioutliers which are corresponding to the activities of control flow. To be specific, firstly, whether the current activity can occur is determined as shown in Step 3.1. If the answer is yes, then determining whether the current activity is specific or not, i.e., T_6 or T_{30}. Our method defines some activities as outliers when certain they happen. It can be seen from Step 3.1.1 of Algorithm 1, we carry out special processing on activities T_6 and T_{30}. That means if the dynamic behavior of the user is abnormal or there exist other unpaid orders besides the current one; then, these activities will be marked as outliers and they are set to 1 in the sequence of abnormalities. In other words, as long as any of these activities occur, they will be regarded as suspicious outliers without judgment. If the current activity is not one of the specific activities, then go to Step 3.1.2. First, whether the activity has corresponding guards will be judged. If it has guards, the variables of activity T will be processed with parameter Φ and others. For

Input: $DPNE=(PN\text{-}DO, V, U, R, W, G, \Phi)$, initial marking M_0; T_x refers to the transition in Table 2, and x is the serial number, $x\in[0, 31]$;
Output: event log L, the outlier sequences of L is stored in B;
1. $L=\varnothing$, $B=\varnothing$;
2. Let M_0 be the root node, and mark it as "control-enabledness and data-enabledness";
3. **While** "control-enabledness and data-enabledness" nodes exist **Do**
Choose the "control-enabledness and data-enabledness" node as M';
3.1 **If** $\exists t \in T \rightarrow M'[_c t >$ and $M'[_d t >$ **Then**
$L=L\cup\{t\}$;
3.1.1 **If** $t=T_6$ or $t=T_{30}$ **Then**
$B[t]=1$;
Go to 3;
End
3.1.2 **Else**
Selected the transition t and determine if it meets the guards;
3.1.2.1 **If** $G(t)! = \varnothing$ **Then**
3.1.2.1.1 **If** $\Phi(t) = \varnothing$ **Then**
3.1.2.1.1.1 **If** t meets $G(t)$ **Then**
$B[t]=0$;
End
3.1.2.1.1.2 **Else**
$B[t]=1$;
Go to 3;
End
End
3.1.2.1.2 **Else**
Get $v(r)$ and $v(w)$ of t as input, and according to the predetermined algorithm of current t, use it as $\Phi(t)$ to process data;
3.1.2.1.2.1 **If** t meets $G(t)$ **Then**
$B[t]=0$;
End
3.1.2.1.2.2 **Else**
$B[t]=1$;
Go to 3;
End
End
End
3.1.2.2 **Else**
$B[t]=0$
Go to 3;
End
End
End
3.2 **Else**
$T=T\text{-}L$; $L=\varnothing$; $B=\varnothing$;
Go to 2;
End
End

Algorithm 1: Outlier extraction.

example, when the current activity t is T_5 i.e., *operation detection*, $\Phi(T_5)$ is the "recursive correlation algorithm". Then, the read variables will be processed by $\Phi(T_5)$; in the same way, $\Phi(T_{28})$ is the "full sequence comparison algorithm." And the read variables of T_{28} will be processed by $\Phi(T_{28})$. Then, the guard function of it is used to judge the value of the variable after processing. If it is normal, it will be marked as 0; and if it is abnormal, it will be marked as 1. For the activity t with no parameter $\Phi(t)$, then, $G(t)$ is directly used to judge its input variables. If the activity has no guard G, it is uniformly marked as 0.

Finally, all the activities are judged, and then, the sequence of abnormal detection results corresponding to the current activity is obtained, which are recorded in dictionary B. In addition, this paper takes e-commerce user behavior anomaly detection as the background. When our method is applied to other scenarios, the parameter $\Phi(t)$ can be changed depending on the scenario.

Input:
1. The outlier sequence set B is composed of the transition set L in Algorithm 1;
2. The set of user's history transaction behavior patterns and the normal patterns B', and its transition sets is represented by L';
3. The weight set of each activity is sequence W, where $W[t_i] = w$;
4. The weight of each normal pattern is sequence Y, where $Y[t_i] = y$;
5. N is the set of all normal behavior patterns B';
6. C is the set of matching results of each activity;

Output: the path L' to be selected by the system.

1. Calculate the matching results between B and B' and store it in Q, initial $Q = \varnothing$;
2. **While** the elements of the set N are not all traversed **Do**
   select $B_j{}'$ from N;
   While the elements of the set L are not all traversed **Do**
   select t_i from L;
   If $B[t_i] == B_j{}'[t_i]$ **Then**
   $Q_j[t_i] = 0$;
   Else
   $Q_j[t_i] = 1$;
   End
End
3. Calculate each matching results C_j under the current normal behavior patterns $B_j{}'$ by formula (2);
4. Calculate the final anomaly detection result S by formula (3);
   4.1 **If** $S \le 0.4$ **Then**
   $L' = (P_{29}, T_{33})$;
   4.2 **Else**
   $L' = (P_{29}, T_{32})$;

Algorithm 2: Abnormal order detection.

$$C_j = \frac{\sum_{i=1}^{n} Q_j[t_j] \times w_i}{\sum_{i=1}^{n} w_i}, \tag{2}$$

$$S = \frac{\sum_{j=1}^{m} C_j[l_j] \times y_j}{m}. \tag{3}$$

Theorem 5. *Algorithm 1 is terminated.*

Proof. Through the structural analysis of the model in the previous section, under any identification, the input of Algorithm 1 can reach the final state space. Algorithm 1 starts from starting place of the model and selects the enabled transition t with M_0. The algorithm judges each transition in the model through the cycle in Step 3. The event log is formed by L which stores the transition path. If there is no enabled transition, indicating that the current activity cannot proceed, set path L empty, then go back to step 2 and reselect the other enabled transition t. This means that the "enabled" marking set is constantly updated until there are no enabled transitions, so it is finite and Algorithm 1 can be terminated.

Algorithm 1 can judge the current behavior of each activity of the e-commerce transaction model from the perspective of control flow and data flow. The path set L of the current transaction and the sequence B of abnormal judgment markers corresponding to each activity can be obtained. Algorithm 1 is to analyse and judge the outliers in the whole trading process. This result will be the input to Algorithm 2. The time complexity of Algorithm 1 is $O(n)$. □

Algorithm 2 is used for processing the data extracted by Algorithm 1. It can be mainly divided into three steps. The first is to match the judgment result sequence of current activity with the standard sequence and the sequence of user history, which is the personalized judgment processing from the perspective of user habit; the second is to assign different parameters according to the severity of different activities, and carry out a weighted average of the matching results of the first step so as to obtain more comprehensive analysis results. Formula (2) is used to process the score of matches between the current sequence and several normal sequences; third, each matching score obtained by the last step is processed by formula (3) to obtain the final result.

By analyzing the matching results of the current transaction pattern and several normal patterns, the final result will be used as the criteria for selection in the process paths between (P_{29}, T_{33}) and (P_{29}, T_{32}).

Theorem 6. *Algorithm 2 is terminated.*

Proof. The number of elements i.e., transition t in the path set L is limited, so it can be all traversed, and the inner loop can be ended. The number of elements i.e., normal behavior patterns B' in set N is also limited, so it can be all traversed. Therefore, the outer loop can be ended. In summary, the two

layers of loops can be ended. It is proved that Algorithm 2 can be terminated. The time complexity of Algorithm 2 is $O(n^2)$. □

5. Case Study

In the previous section, the relevant data processing algorithms are introduced. In this section, we use a set of simulated event logs and data sets to replay and match the above models using Algorithm 1. As a result, a series of outliers can be obtained. Algorithm 2 is then used to determine whether the current transaction is normal or abnormal.

We analyse and judge the process data of successful transactions but fail to establish order information for the process and data of failed transactions, so we do not analyse them. For example, we now have a sequence of activities as follows:

$$(T_0, T_1, T_2, T_1, T_3, T_4, T_5, T_7, T_8, T_9, T_{12}, T_{13}, T_{14}, T_{16}, T_{17}, T_{15}, T_{19}, T_{20}, T_{22}, T_{21}, T_{23}, T_{24}, T_{26}, T_{27}, T_{25}, T_{28}, T_{30}, T_{29}). \tag{4}$$

In the standard schema, the normal schema defined by the expert is a set of 0 s. There will be some specific habits in the consumption process of buyers. Even if these habits cannot be strictly implemented under the rules defined by experts, they also have a certain reference value, which can reduce the misjudgment of users. The set of assumed normal patterns and patterns that represent the user's current normal habits is shown below:

$$\begin{aligned}&(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),\\&(0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,0),\\&(0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,1,0),\\&(0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0).\end{aligned} \tag{5}$$

Assume that the weight set of each activity is sequence W defined by the expert is (0, 0, 2, 0, 2, 0, 0, 6, 0, 2, 0, 5, 5, 2, 0, 2, 1, 0, 0, 3, 3, 3, 3, 3, 3, 4, 3, 0). The weight of each normal pattern sequence Y is 40%, 20%, 20%, and 20%, respectively. In some cases, if all the current behaviors are normal, i.e., the above sequence of activities is the same as the corresponding values defined by the expert. According to Algorithm 1, the matching result between the current activity and the preset activity can be obtained. According to Algorithm 2 and the weights which are given above, we can get an anomaly probability of about 0.115. In the same way, if all the current behaviors are abnormal, then the abnormal possibility is about 0.885.

We operate the following bindings with Algorithms 1 and 2 in turn to obtain the results of anomaly analysis after considering the three most recent patterns in the user's history.

$$\delta_E = \left\langle \begin{array}{l} (T_0, \{V_0, ID\_12345\}, \{V_1, 2\}, \varnothing), \\ (T_1, \{V_1, 2\}, \varnothing, \varnothing) \\ (T_2, \varnothing, \{V_1, 3\}, \varnothing) \\ (T_1, \{V_1, 3\}, \varnothing, \varnothing) \\ (T_3, \varnothing, \varnothing, \varnothing) \\ (T_4, \varnothing, \{V_2, 121212134612414567\}, \varnothing) \\ (T_5, \{\{V_2, 121212134612414567\}, \{V_2', ...\}\}, \{V_3, 0.2\}, recursive\_correlation) \\ (T_7, \{V_3, 0.2\}, \{V_4, Goods\_\inf o(134, 25, 66)\}, \varnothing) \\ \left(T_8, \{V_4, Goods\_\inf o(134, 25, 66)\}, \left\{\begin{matrix}\{V_5, TRUE\}, \\ \{V_6, FALSE\}\end{matrix}\right\}, \varnothing\right) \\ \left(T_9, \left\{\begin{matrix}\{V_5, TRUE\}, \\ \{V_6, FALSE\}\end{matrix}\right\}, \{\{V_{10}, TRUE\}, \{V_{11}, B\_110\}\}, \varnothing\right) \\ (T_{12}, \varnothing, \{V_7, TRUE\}, \varnothing) \\ (T_{13}, \{V_7, TRUE\}, \{V_8, \text{TRUE}\}, \varnothing) \\ (T_{14}, \{\{V_8, \text{TRUE}\}, \{V_{10}, TRUE\}\}, \{V_9, \#abc890\}, \varnothing) \\ ..... \end{array} \right\rangle \tag{6}$$

As shown in the above cases, the order is normal only from the perspective of control flows (including P_c and P_d). After considering data attributes, some of the activities are abnormal. Then, the case is analysed, and it is found that the anomaly probability is about 0.415. Because it is greater than the defined threshold of 0.4, the activity T_{32} will be chosen. That is, the order is classified as an abnormal order and the background will take steps to block this transaction. Thus, our e-commerce transaction model and abnormal order detection algorithm can judge the event log sequence and

database and monitor the transaction process timely, as well as judging and intercepting the current transaction in real time.

6. Conclusions

For the good development of e-commerce, a secure transaction environment is needed as an important guarantee. Based on this research background, this paper is aimed at proposing an outlier analysis method based on the formal model and process system. A successful process system relies on effective modeling and analysis. Both control flows and data flows should be considered [27]. To further expand its data detection and analysis capabilities, this paper proposes DPNE, which is based on PN-DO and DPN. It introduces algorithms and functions into activities so that the conformance detection function can be extended. Based on the e-commerce model shown as DPNE which includes mobile transaction process, the relevant outlier extraction and anomaly orders detection algorithm are proposed. Through the algorithm, the current transaction data analysis could be used to determine whether the current transaction and the user are abnormal. Then, the data analysis technology that can analyse the anomaly of e-commerce transaction flow from the perspective of process and data is established.

In conclusion, this paper mainly carries out the construction and analysis of the transaction model and realizes some basic functions of outlier capture. The methods in this paper can also be applied to other areas, such as social networks [43]. In the future work, we will do further research on expanding the types of anomalous trades and develop more efficient algorithms to detect the anomaly of e-commerce transactions.

Data Availability

The simulated data used to support the findings of this study are available upon request to the first author.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work was supported in part by the Natural Science Foundation of Shaanxi Province under Grant 2021JM-205 and by the Fundamental Research Funds for the Central Universities of China under Grant GK202003080.

References

[1] S. Fatonah, A. Yulandari, and F. W. Wibowo, "A review of e-payment system in e-commerce," *Journal of Physics: Conference Series. IOP Publishing*, vol. 1140, no. 1, article 012033, 2018.

[2] M. F. Putri, B. Purwandari, and A. N. Hidayanto, "What do affect customers to use mobile payment continually? A systematic literature review," in *Fifth International Conference on Informatics and Computing (ICIC)*, pp. 1–6, Gorontalo, Indonesia, 2020.

[3] S.-H. Chun, "E-commerce liability and security breaches in mobile payment for e-business sustainability," *Sustainability*, vol. 11, no. 3, p. 715, 2019.

[4] F. Hao, H. Guo, D.-S. Park, and J. Kang, "An efficient pricing strategy of sensing tasks for crowdphotographing," *The Journal of Supercomputing*, vol. 75, no. 8, pp. 4443–4458, 2019.

[5] A. Abdallah, M. A. Maarof, and A. Zainal, "*Fraud detection system: a survey*," *Journal of Network and Computer Applications*, vol. 68, pp. 90–113, 2016.

[6] Y. Wang, K. Streff, and S. Raman, "Smartphone security challenges," *IEEE Computer Architecture Letters*, vol. 45, no. 12, pp. 52–58, 2012.

[7] R. Wang, S. Chen, X. Wang, and S. Qadeer, "How to shop for free online–security analysis of cashier-as-a-service based web stores," in *2011 IEEE symposium on security and privacy*, pp. 465–480, Oakland, CA, USA, 2011.

[8] W. Yang, Y. Zhang, J. Li et al., *Show me the money! Finding flawed implementations of third-party in-app payment in android apps*, NDSS, 2017.

[9] B. Baesens, S. Höppner, and T. Verdonck, "Data engineering for fraud detection," *Decision Support Systems*, vol. 150, p. 113492, 2021.

[10] L. Zheng, G. Liu, C. Yan, C. Jiang, M. Zhou, and M. Li, "Improved TrAdaBoost and its application to transaction fraud detection," *IEEE Transactions on Computational Social Systems*, vol. 7, no. 5, pp. 1304–1316, 2020.

[11] W. Van Der Aalst, "Process mining," *Communications of the ACM*, vol. 55, no. 8, pp. 76–83, 2012.

[12] F. Hao, E. Yang, L. Guo, A. Nasridinov, and D.-S. Park, "On invariance of concept stability for attribute reduction in concept lattice," in *Advances in Computer Science and Ubiquitous Computing*, pp. 101–106, Springer, Singapore, 2021.

[13] W. Y. Yu, C. G. Yan, Z. J. Ding, C. J. Jiang, and M. C. Zhou, "Modeling and validating e-commerce business process based on Petri nets," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 44, no. 3, pp. 327–341, 2014.

[14] Y. Y. Du, L. Qi, and M. C. Zhou, "Analysis and application of logical Petri nets to E-commerce systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 44, no. 4, pp. 468–481, 2014.

[15] W. Liu, X. Feng, F. Zhang, Y. du, and C. Yan, "Analytic of B2C E-commerce credit mechanism mixed strategy risk behavior based on logical game Petri nets," *IEEE Access*, vol. 6, pp. 29109–29131, 2018.

[16] D. Xiang, G. Liu, C. Yan, and C. Jiang, "Checking the inconsistent data in concurrent systems by Petri nets with data operations," in *2016 IEEE 22nd International Conference on Parallel and Distributed Systems (ICPADS)*, pp. 501–508, Wuhan, China, 2016.

[17] M. De Leoni and W. M. van der Aalst, "Data-aware process mining: discovering decisions in processes using alignments," in *Proceedings of the 28th annual ACM symposium on applied computing*, pp. 1454–1461, Coimbra, Portugal, 2013.

[18] T. Murata, "Petri nets: properties, analysis and applications," *Proceedings of the IEEE*, vol. 77, no. 4, pp. 541–580, 1989.

[19] Y.-F. Lian, "Anomaly detection of user behaviors based on profile mining," *Chinese Journal of Computers-Chinese Edition*, vol. 25, no. 3, pp. 325–330, 2002.

[20] J. Hopcroft and R. Tarjan, "Algorithm 447: efficient algorithms for graph manipulation," *Communications of the ACM*, vol. 16, no. 6, pp. 372–378, 1973.

[21] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection," *ACM Computing Surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009.

[22] L. Zheng, G. Liu, C. Yan, and C. Jiang, "Transaction fraud detection based on total order relation and behavior diversity," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 3, pp. 796–806, 2018.

[23] H. Weng, Z. Li, S. Ji et al., "Online e-commerce fraud: a large-scale detection and analysis," in *2018 IEEE 34th International Conference on Data Engineering (ICDE)*, pp. 1435–1440, Paris, France, 2018.

[24] J. Nanduri, Y. Jia, A. Oka, J. Beaver, and Y.-W. Liu, "Microsoft uses machine learning and optimization to reduce E-commerce fraud," *INFORMS Journal on Applied Analytics*, vol. 50, no. 1, pp. 64–79, 2020.

[25] L. Vanneschi, D. M. Horn, M. Castelli, and A. Popovič, "An artificial intelligence system for predicting customer default in e-commerce," *Expert Systems with Applications*, vol. 104, pp. 1–21, 2018.

[26] W. Yu, C. Yan, Z. Ding, C. Jiang, and M. Zhou, "Analyzing E-commerce business process nets via incidence matrix and reduction," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 1, pp. 130–141, 2018.

[27] W. Yu, Z. Ding, L. Liu, X. Wang, and R. D. Crossley, "Petri net-based methods for analyzing structural security in e-commerce business processes," *Future Generation Computer Systems*, vol. 109, pp. 611–620, 2020.

[28] W. Yu, L. Liu, Y. An, and X. Zhai, "Analyzing the validation flaws of online shopping systems based on coloured Petri nets," in *2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, pp. 1706–1710, Leicester, UK, 2019.

[29] B. Jasiul, M. Szpyrka, and J. Śliwa, "Detection and modeling of cyber attacks with petri nets," *Entropy*, vol. 16, no. 12, pp. 6602–6623, 2014.

[30] D. Xiang, G. Liu, C. Yan, and C. Jiang, "Detecting data-flow errors based on Petri nets with data operations," *IEEE/CAA Journal of Automatica Sinica*, vol. 5, no. 1, pp. 251–260, 2018.

[31] A. Rozinat and W. M. P. Van der Aalst, "Conformance checking of processes based on monitoring real behavior," *Information Systems*, vol. 33, no. 1, pp. 64–95, 2008.

[32] M. de Leoni, J. Munoz-Gama, J. Carmona, and W. M. P. van der Aalst, "Decomposing alignment-based conformance checking of data-aware process models," in *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pp. 3–20, Springer, Berlin, Heidelberg, 2014.

[33] A. Adriansyah, B. F. van Dongen, and W. M. P. van der Aalst, "Conformance checking using cost-based fitness analysis," in *2011 Ieee 15th international enterprise distributed object computing conference*, pp. 55–64, Helsinki, Finland, 2011.

[34] W. M. P. van der Aalst, *Business Alignment: Using Process Mining as a Tool for Delta Analysis*, CAiSE Workshops, 2004.

[35] S. Haarmann, K. Batoulis, and M. Weske, "Compliance checking for decision-aware process models," in *International Conference on Business Process Management*, pp. 494–506, Springer, Cham, 2019.

[36] A. Rozinat and W. M. P. van der Aalst, "Decision mining in ProM," in *International Conference on Business Process Management*, pp. 420–425, Springer, Berlin, Heidelberg, 2006.

[37] R. Sarno, P. L. I. Sari, H. Ginardi, D. Sunaryono, and I. Mukhlash, "Decision mining for multi choice workflow patterns," in *2013 International conference on computer, control, informatics and its applications (IC3INA)*, pp. 337–342, Jakarta, Indonesia, 2013.

[38] Y. Wang, C. Hahn, and K. Sutrave, "Mobile payment security, threats, and challenges," in *2016 Second international conference on mobile and secure services (MobiSecServ)*, pp. 1–5, Gainesville, FL, USA, 2016.

[39] J. Wang, *Timed Petri Nets: Theory and Application*, Springer Science & Business Media, 2012.

[40] C. Haoxun, "Net structure and control logic synthesis of controlled Petri nets," *IEEE Transactions on Automatic Control*, vol. 43, no. 10, pp. 1446–1450, 1998.

[41] W. van der Aalst, A. Adriansyah, and B. van Dongen, "Replaying history on process models for conformance checking and performance *analysis*," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 182–192, 2012.

[42] J. I. Bing-Shuai, L. I. Hu, W. H. Han, and Y. Jia, *Research on e-commerce-oriented user abnormal behaviour detection*, NetinfoSecurity, 2014.

[43] F. Hao, J. Gao, J. Chen, A. Nasridinov, and G. Min, "Skyline (λ, k)-cliques identification from fuzzy attributed social networks," *IEEE Transactions on Computational Social Systems*, pp. 1–12, 2021.