World Wide Web (2013) 16:449–475
DOI 10.1007/s11280-012-0178-0

Effective detection of sophisticated online banking fraud on extremely imbalanced data

Wei Wei · Jinjiu Li · Longbing Cao · Yuming Ou · Jiahang Chen

Received: 10 January 2012 / Revised: 25 April 2012 /
Accepted: 25 June 2012 / Published online: 19 July 2012

Abstract Sophisticated online banking fraud reflects the integrative abuse of resources in social, cyber and physical worlds. Its detection is a typical use case of the broad-based Wisdom Web of Things (W2T) methodology. However, there is very limited information available to distinguish dynamic fraud from genuine customer behavior in such an extremely sparse and imbalanced data environment, which makes the instant and effective detection become more and more important and challenging. In this paper, we propose an effective online banking fraud detection framework that synthesizes relevant resources and incorporates several advanced data mining techniques. By building a contrast vector for each transaction based on its customer's historical behavior sequence, we profile the differentiating rate of each current transaction against the customer's behavior preference. A novel algorithm, ContrastMiner, is introduced to efficiently mine contrast patterns and distinguish fraudulent from genuine behavior, followed by an effective pattern selection and risk scoring that combines predictions from different models. Results from experiments on large-scale real online banking data demonstrate that our system can achieve substantially higher accuracy and lower alert volume than the latest benchmarking fraud detection system incorporating domain knowledge and traditional fraud detection methods.

W. Wei · J. Li · L. Cao (✉) · Y. Ou · J. Chen
Advanced Analytics Institute, University of Technology Sydney, Sydney, Australia
e-mail: longbing.cao@uts.edu.au

W. Wei
e-mail: Wei.Wei-7@student.uts.edu.au

J. Li
e-mail: Jinjiu.Li@eng.uts.edu.au

Y. Ou
e-mail: yuming.ou@uts.edu.au

J. Chen
e-mail: chen_jiahang@hotmail.com

Keywords fraud detection · online banking · contrast pattern · neural network · data mining

1 Introduction

With the widespread use of increasingly advanced Internet technology [15, 47, 60], online banking (also called Internet banking) is emerging as a major channel for retail and business banking. In contrast, fraudulent online banking activities are becoming more and more sophisticated, seriously threatening the security and trust of online banking business. Online banking fraud has become a serious issue in financial crime management for all banks. It is becoming ever more challenging and leads to massive losses, due to the emergence and evolution of sophisticated online banking fraud, such as phishing scams, malware infection and ghost web sites. Effective and efficient detection of Internet banking fraud is regarded as a major challenge to all banks, and is an increasing cause for concern.

An online banking fraud detection system can be a typical use case of the broad-based Wisdom Web of Things (W2T) [63–66] methodology. It has to timely gather multi-aspect data of online banking customers, including demographic data, online banking transaction data, credit card transaction data and other types of transaction data.These data will be transferred via the Internet/WWW and SEA-nets to an online banking customer data center. The data center provides a platform for the whole process of online banking fraud detection. It is a complete data cycle from acquisition of heterogeneous data, information, and knowledge in the physical world to the provision of active services in the cyber world to customers in the social world. Online banking customers (in the social world), things (in the physical world), and computer systems (in the cyber world) are integrated into an entity to realize their harmony and symbiosis by using an effective W2T data cycle. In this cycle, the process of fraud detection is one important task.

Internet banking fraud exhibits certain sophisticated characteristics (see detailed discussions in Section 2.1):

- suspicious customers are active and intelligent in conducting fraudulent banking activities,
- fraudulent behavior is very dynamic,
- fraud is hidden in diversified customer behavior,
- fraud-related transactions are dispersed in highly imbalanced large data sets, and
- the occurrences of fraud appear in a very limited time which requires real-time detection.

The detection of online banking fraud needs to be instant, because it is very difficult to recover the loss if a fraud is undiscovered during the detection period. Most customers usually rarely check their online banking history regularly and are therefore not able to discover and report fraud transactions immediately after an occurrence of a fraud. This makes the possibility of loss recovery very low. In addition, all alerts generated from the detection system need to be manually investigated, which is very time-consuming. Online banking detection systems are therefore expected to have high accuracy, a high detection rate, and a low false positive rate for generating a small, manageable number of alerts in complex online banking business.

The above characteristics and business requirements greatly challenge existing fraud detection techniques and data mining models for protecting credit card transactions, e-commerce, insurance, retail, telecommunication, computer intrusion, etc. These existing methods demonstrate poor performance in efficiency and/or accuracy when directly applied to online banking fraud detection [35]. For instance, credit card or telecommunication fraud detection often focuses on discovering particular behavior patterns of a specific customer or group, but fraud-related online banking transactions are very dynamic and appear very similar to genuine customer behavior. Some intrusion detection methods perform well in a dynamic computer environment, but they require a large amount of training data with complete attack logs as evidence. However, there is no obvious evidence to show whether an online banking transaction is fraudulent.

A promising direction emerged recently that scrutinizes the difference between fraudulent and genuine behavior, and develops corresponding approaches for mining contrast patterns, for instance, contrast sets [6] and emerging patterns [24, 25, 52]. However, experiments of classic methods on real online banking data have shown that their accuracy is not very high because of the challenges in online banking fraud detection. In addition, according to the research in [61], contrast pattern mining is an NP hard problem, the time cost is expensive, especially when the number of attributes is large, and the threshold of minimal detection rate is small. Based on our experiments, the contrast pattern method in [24] does not perform efficiently in the online banking scenario.

There are few papers about fraud control in online banking [35, 37, 44]. The mainstream online banking fraud detection systems rely on domain experts and knowledge to create rules for filtering suspicious transactions, which face critical problems, including very high false positive rates and low detection rates. More importantly, the adaptation of rules to fraud dynamics is fully dependent on domain expertise. This is very time-consuming, leaves the quality of fraud detection without sustainable control, and cannot support instant adjustment of rules.

Most previous work treats events at different time points as independent and ignores the information incorporated in event sequences. In online banking, activity sequences are useful for differentiating fraudulent behavior from genuine behavior. An example is shown in Tables 1 and 2. Table 1 is a web page access sequence committed by a Trojan, while Table 2 is from a genuine transaction via a web browser. There are two contrasting features between these two sequences. One is that the fraud bypassed some web pages that are insignificant for submission of the transaction, such as homepage.aspx after login and the print page after the transfer confirmation. The other is that the transaction was completed within 3 seconds of login, which is too fast for a common online banking user to achieve via a web browser.

Table 1 Fraud behavior sequence.

Time	PageLink
21:55:42.190	Login.aspx
21:55:43.260	BalanceCheck.aspx
21:55:43.890	PayForm.aspx
21:55:44.121	PayConfirm.aspx
21:55:45.091	HomePage.aspx

Table 2 Genuine behavior sequence.

Time	PageLink
21:58:06.190	Login.aspx
21:58:07.391	HomePage.aspx
21:58:15.260	BalanceCheck.aspx
21:58:27.890	PayForm.aspx
21:59:22.121	PayConfirm.aspx
21:59:27.091	Print.aspx
21:59:32.091	HomePage.aspx

Using the above data and business characteristics, this paper proposes an effective framework for detecting sophisticated Internet banking fraud efficiently. The main ideas, advantages and resulting contributions of this framework are as follows:

- It is inspired by the theory of meta-synthetic engineering [11], M-Computing [12] and Wisdom Web of Things [66], and provides a systematic solution by synthesizing domain knowledge, experience learned in the rule-based detection system, advantages from multiple models, and refinement by domain experts.
- It embeds systematic modules by selecting features based on information gain, extracting contrast behavior, building classifiers, generating an overall risk score for every online banking transaction, and identifying patterns of fraudulent behavior. This makes it a real time online banking fraud detection system that does not interfere with any existing online banking system or its service.
- We not only construct sequence behavior information for identifying contrast patterns, but also propose a new method, a contrast vector, to integrate the sequential behavior contrast into the relational transaction database for mining more effective contrast patterns.
- The system incorporates and integrates several data mining models, cost-sensitive neural network [67], contrast pattern mining, and decision forest. Because different models discover fraud and genuine behavior patterns from different angles, their combination [13] captures behavior patterns in a more comprehensive way.
- Each model can be easily retrained over time to keep abreast of changes in fraud behavior.
- Massive experiments in a major Australian bank show that our system and models have a higher detection rate and a lower false positive rate than any single classic data mining model, outperforming the existing rule-based system used in all major Australian banks. In addition, our system generates comparably good detection performance on highly imbalanced data sets and the modified contrast pattern mining model is efficient on real time data. The sequence behavior patterns discovered also provide more information about forensic evidence for fraud detection.

The remainder of the paper is organized as follows. Section 2 describes the characteristics of online banking fraud in detail and presents an overview of related work on fraud detection. Section 3 gives a problem statement and definition of terminology, while Section 4 presents and explains the online banking fraud detection framework in detail. The method of contrast pattern mining with contrast vectors is introduced in Section 5 and the risk scoring method based on combined models

is presented in Section 6. Experiment evaluation is discussed in Sections 7 and 8 provides conclusions and suggests future research directions.

2 Online banking fraud characteristics and related work

In this section, we first summarize the main characteristics of online banking fraud, and then discuss the related work on different areas of fraud detection. Most published work about fraud detection is related to the domain of credit card fraud, computer intrusion and telecommunication fraud. We therefore discuss each of these and explain the limitations of the existing work when applied to detect online banking fraud.

2.1 Online banking fraud characteristics

From a system point of view, the essence of online fraud reflects the synthetic abuse of interaction between resources in three worlds: the fraudster's intelligence abuse in the social world, the abuse of web technology and Internet banking resources in the cyber world, and the abuse of trading tools and resources in the physical world. This is a typical example of a problem in the Wisdom Web of Things (W2T). A close investigation of the characteristics is important for developing effective solutions, which will then be helpful for other problem-solving in W2T.

Our investigations in one of the largest banks in Australia show that real-world online banking transaction data sets and most online banking fraud has the following characteristics and challenges: (1) highly imbalanced large data set; (2) real time detection; (3) dynamic fraud behavior; (4) weak forensic evidence; and (5) diverse genuine behavior patterns.

(1) *The data set is large and highly imbalanced.* According to our study on one Australian bank's online banking data, online banking fraud detection involves a large number of transactions, usually millions. However, the number of daily frauds is usually very small. For instance, there were only 5 frauds among more than 300,000 transactions on one day. This results in the task of detecting very rare fraud dispersed among a massive number of genuine transactions.

(2) *Fraud detection needs to be real time.* In online banking, the interval between a customer making a payment and the payment being transferred to its destination account is usually very short. To prevent instant money loss, a fraud detection alert should be generated as quickly as possible. This requires a high level of efficiency in detecting fraud in large and imbalanced data.

(3) *The fraud behavior is dynamic.* Fraudsters continually advance their techniques to defeat online banking defenses. Malware, which accounts for the greater part of online banking fraud, has been reported to have over 55,000 new malicious programs everyday [5]. This puts fraud detection in the position of having to defend against an ever-growing set of attacks. This is far beyond the capability of any single fraud detection model, and requires the adaptive capability of models and the possibility of engaging multiple models [13] for leveraging the challenges that cannot be handled by any single model.

(4) *The forensic evidence for fraud detection is weak.* For online banking transactions, it is only possible to know source accounts, destination accounts and

dollar value associated with each transaction, but other external information, for example, the purpose of the spending, is not available. Moreover, with the exception of ID theft, most online banking fraud is not caused by the hijack of an online banking system but by attacks on customers' computers. In fraud detection, only the online banking activities recorded in banking systems can be accessed, not the whole compromise process and solid forensic evidence (including labels showing whether a transaction is fraudulent) which could be very useful for understanding nature of the deception. This makes it challenging to identify sophisticated fraud with very limited information.

(5) *The customer behavior patterns are diverse.* An online banking interface provides a one-stop entry for customers to access most banking services and multiple accounts. In conducting online banking business, every customer may perform very differently for different purposes. This leads to a diversity of genuine customer transactions. In addition, fraudsters simulate genuine customer behavior and change their behavior frequently to compete with advances in fraud detection. This makes it difficult to characterize fraud and even more difficult to distinguish it from genuine behavior.

(6) *The online banking system is fixed.* The online banking process and system of any bank are fixed. Every customer accesses the same banking system and can only use the services in a predefined way. This leads to good references for characterizing common genuine behavior sequences, and for identifying tiny suspicions in fraudulent online banking.

The above characteristics make it very difficult to detect online banking fraud, and online banking fraud detection presents several major challenges to the WWW research and W2T, especially for the mainstream data mining community: extremely imbalanced data, big data, model efficiency in dealing with complex data, dynamic data mining, pattern mining with limited or no labels, and discriminant analysis of data without clear differentiation. In addition, it is very challenging to develop a single model to tackle all of the above aspects, which greatly challenge the existing work in fraud detection.

2.2 General work in fraud detection

Many statistic and machine learning techniques have been developed for tackling fraud [54], for example, Neural Network, Decision Tree [48], Logistic Regression [4] and Rule-based Expert Systems [22]. They have been used to detect abnormal activities and for fraud detection in many fields, such as money laundering, credit card fraud, computer intrusion [29], and so on. They can be categorized as unsupervised approaches and supervised ones. Unsupervised approaches, such as Hidden Markov Model [46, 56], are mainly used in outlier detection and spike detection when the training samples are unlabeled. Based on historical data and domain knowledge, online banking can collect clearly labeled data samples for the reports from victims or related crime control organizations. Unsupervised approaches cannot use such label information, and the accuracy is lower than that of supervised approaches. Some supervised methods, such as Neural Network and Random Forests [10], perform well in many classification applications, including fraud detection applications, even in certain class-imbalanced scenarios [2, 9, 14, 42, 50, 67]. However, they either cannot

tackle extremely imbalanced data, or are not capable of dealing with comprehensive complexities as shown in the online banking data and business.

Understanding the complexities of contrast between fraudulent behavior and genuine behavior can also provide essential patterns which, when incorporated in a classifier, lead to high accuracy and predictive power. Such understanding triggers the emergence of contrast pattern mining, such as emerging pattern [24, 25], jumping emerging patterns [41], and mining contrast sets [6]. However, our experiments show that these approaches are not efficient for detecting rare fraud among an extremely large number of genuine transactions.

2.3 Fraud detection in online banking

There are very few papers about fraud detection in online banking [49]. Most of them concern fraud prevention, which uses efficient security measures to prevent fraudulent financial transactions performed by unauthorized users and to ensure transaction integrity [8, 20, 28, 33, 39]. Aggelis [1] proposed an online banking fraud detection system for offline processing. Another system presented in [37] works well online but needs a component that must be downloaded and installed in the client device, which is inconvenient for deployment.

In practice, typical existing online banking fraud detection systems are rule based and match likely fraud in transactions. The rules are mostly generated according to domain knowledge; consequently, these systems usually have a high false positive rate but a low fraud detection rate. Importantly, the rules are not adaptive to changes in the types of fraud.

2.4 Credit card fraud detection

Credit card fraud is divided into two types: offline fraud and online fraud. Offline fraud is committed by using a stolen physical card at a storefront or call center. In most cases, the institution issuing the card can lock it before it is used in a fraudulent manner, if the theft is discovered quickly enough. Online fraud is committed via web, phone shopping or cardholder-not-present. Only the card's details are needed, and a manual signature and card imprint are not required at the time of purchase. With the increase of e-commence, online credit card transaction fraud is increasing. Compared to online banking fraud detection, there are many available research discussions and solutions about credit card fraud detection [3, 23, 43].

Most of the work on preventing and detecting credit card fraud has been carried out with neural networks [36]. CARDWATCH [2] features a neural network trained with the past data of a particular customer and causes the network to process current spending patterns to detect possible anomalies. Brause and Langsdorf proposed a rule-based association system combined with the neuro-adaptive approach [9]. Falcon, developed by HNC, uses feed-forward Artificial Neural Networks trained on a variant of a back-propagation training algorithm [32]. Machine learning, adaptive pattern recognition, neural networks, and statistical modeling are employed to develop Falcon predictive models to provide a measure of certainty about whether a particular transaction is fraudulent. A neural MLP-based classifier is another example of a system that uses neural networks [26]. It acts only on the information of the operation itself and of its immediate previous history, but not on historic databases of

past cardholder activities. A parallel Granular Neural Network (GNN) method uses a fuzzy neural network and rule-based approach [57]. The neural system is trained in parallel using training data sets, and the trained parallel fuzzy neural network then discovers fuzzy rules for future prediction. CyberSource introduces a hybrid model, combining an expert system with a neural network to increase its statistic modeling and reduce the number of "false" rejections [19]. There are also some unsupervised methods, such HMM [56] and cluster [46], targeting unlabeled data sets.

All credit card fraud detection methods seek to discover spending patterns based on the historical data of a particular customer's past activities. It is not suitable for online banking because of the diversity of online banking customers' activities and the limited historical data available for a single customer.

2.5 Computer intrusion detection

Many intrusion detection systems base their operations on analysis of audit data generated by the operation system. Intrusion detection approaches in computers are broadly classified into two categories based on a model of intrusions: misuse and anomaly detection. Misuse detection attempts to recognize the attacks of previously observed intrusions in the form of a pattern or signature, and then monitors such occurrences [30, 34, 38]. Misuse approaches include expert systems, model-based reasoning, state transition analysis, and keystroke dynamics monitoring [58]. Misuse detection is simple and fast. Its primary drawback is that it is not possible to anticipate all the different attacks because it looks for only known patterns of abuse. Anomaly detection tries to establish a historical normal profile for each user and then uses a sufficiently large deviation from the profile to indicate possible intrusions [30, 55]. Anomaly detection approaches include statistical approaches, predictive patten generation, and neural networks. The advantage of anomaly detection is that it is possible to detect novel attacks; its weakness is that it is likely to have high rates of false alarm.

Data mining approaches can be applied for intrusion detection. A classification model with association rules algorithm and frequent episodes has been developed for anomaly intrusion detection [40]. This approach can automatically generate concise and accurate detection models from a large amount of audit data. However, it requires a large amount of audit data in order to compute the profile rule sets. Because most forensic evidence for fraud is left on customers' computers and it is difficult to retrieve, intrusion detection methods cannot be directly used for online banking.

2.6 Telecommunication fraud detection

The various types of telecommunication fraud can be classified into two categories: subscription fraud and superimposed fraud. Subscription fraud occurs when a subscription to a service is obtained, often with false identity details and no intention of making payment. Superimposed fraud occurs when a service is used without necessary authority and are usually detected by the appearance of unknown calls on a bill. Research work in telecommunication fraud detection has concentrated mainly on identifying superimposed fraud. Most techniques use Call Detail Record data to create behavior profiles for customers, and detect deviations from these profiles.

Proposed approaches include the rule-based approach [53], neural networks [45, 59], visualization methods [18], and so on. Among them, neural networks can actually calculate user profiles in an independent manner, thus adapting more elegantly to the behavior of various users. Neural networks are claimed to substantially reduce operation costs. As with credit card fraud detection, it is difficult for telecommunication fraud detection methods to characterize the behavior patterns of online banking customers effectively.

Clearly, no single existing method can solve the online banking fraud detection problem easily. Because different approaches have advantages in different aspects, it is believed that a combined solution will outperform any single solution. Neural network has been successfully adopted in all three kinds of fraud detection and is believed to be a stable model. As the online banking behavior sequence data is available from the online banking interface log and is discriminative between abnormal and normal activities, sequential behavior pattern should be included for fraud detection.

3 Problem formulation

In this section, we define concepts and notations that will be used in the paper.

Definition 1 (Transaction) A **transaction** τ is a tuple $\tau = \{a_1, a_2, \cdots, a_L\}$, which is composed of values of all L attributes $\{A_1, A_2, \cdots, A_L\}$ available in banking transactional data. The length of a transaction $|\tau|$ is defined as the number of involved attributes, i.e., $|\tau| = L$.

For instance, a set $\tau_0 = \{age = 25, gender = \text{'}F\text{'}, career = \text{'}student\text{'}\}$ is a transaction with the length $|\tau_0| = 3$. With the concept of transaction, we will convert the original source data to a data set following the above specification. All transactions form a transaction set T and every transaction has a class C ($C \in \{Fraud, Genuine\}$).

Definition 2 (Pattern) A **pattern** P is a combination of attributes, threshold values, and connection operators ($\wedge$). For example, $P_0 = (age \in [25, 35]) \wedge (gender = M) \wedge (career = IT)$ is a pattern with the length $|P_0| = 3$.

Definition 3 (Alert) If a transaction τ satisfies a pattern P, we denote the relationship as $\tau \vDash P$. Accordingly, an **alert** will be triggered on τ.

A transaction set T can be partitioned into two groups regarding to alerts, namely $T_{\vDash}^{(\mathbf{P})}$ and $T_{\nvDash}^{(\mathbf{P})}$, where $T_{\vDash}^{(\mathbf{P})}$ is the set of transactions alerted by the pattern P, and $T_{\nvDash}^{(\mathbf{P})}$ is the set of remaining transactions. Thus, we have:

$$TP^{(\mathbf{P})} = T_{\vDash}^{(\mathbf{P})} \cap T_+, \quad FN^{(\mathbf{P})} = T_{\nvDash}^{(\mathbf{P})} \cap T_+ \tag{3.1}$$

$$FP^{(\mathbf{P})} = T_{\vDash}^{(\mathbf{P})} \cap T_-, \quad TN^{(\mathbf{P})} = T_{\nvDash}^{(\mathbf{P})} \cap T_- \tag{3.2}$$

Here, TP, the true positive number, represents the number of frauds caught by pattern P; FP, the false positive number, denotes the false alerts triggered by P; FN, the false negative number, represents the fraud missed by P; TN, the true

negative number, stands for the genuine transaction number predicted by P. T_+ is the fraud transaction set while T_- is the genuine transaction set, where $T = T_- + T_+$. The objective of fraud detection for online banking is to achieve a higher TP and lower FP.

Definition 4 (Session) A **session** is the period of one customer's behavior between logging in and logging out of the online banking system.

Definition 5 (Event) Let $R = \{A_1, .., A_M\}$ be the full set of event attributes, M is the number of all attributes involved by events. Then an **event** e_t at time t is a $(M + 3)$ tuple: $e_t = ([a_{1t}, ..., a_{Mt}], c, t, s)$, where a_{mt} is the value of attribute A_m at time t, c is the event type, and s is an integer to identify the session number in which e_t occurred. In online banking, there are a variety of event types, and transaction is one type of events.

In one session, there are usually multiple events.

Definition 6 (Event sequence) An **event sequence** $\mathbb{S}$ is a collection of events $\langle e_1, e_2, ..., e_N \rangle$, where N is the number of events in a sequence, $e_n.t < e_{n+1}.t$ ($e.t$ represents an event's time), and events are arranged in the ascending order as per their occurrence time. If events e_i, e_j and e_k occur in the same session, then $e_i.s = e_j.s = e_k.s$ ($e.s$ represents the session of an event). The length of the sequence $\mathbb{S}$ is denoted by $|\mathbb{S}| = N$.

In online banking, sequences reflect customer behavior since they opened their accounts with a bank. A customer's behavior is represented by one sequence.

Table 3 is the activity sequence of a customer. There are two kinds of events: Login and Pay. Each event has its own attributes. e_1, e_4 and e_7 are Login events, and they only involve one attribute A_3, while the rest Pay events involve attributes A_1 and A_2. Some events occur in one session, session s_1 consists of events e_1, e_2 and e_3, and session s_2 involves e_4, e_5 and e_6.

Definition 7 (Sequence database) A **sequence database** is a collection of sequences.

In order to measure the contrast between the current transaction and the transaction history of its customer, we define the contrast vector.

Table 3 Sequence of a customer.

Event id	A_1	A_2	A_3	Event type	Time stamp	Session
e_1	–	–	A	Login	t_1	s_1
e_2	2	10	–	Pay	t_2	s_1
e_3	2	10	–	Pay	t_3	s_1
e_4	–	–	A	Login	t_4	s_2
e_5	2	10	–	Pay	t_5	s_2
e_6	1	5	–	Pay	t_6	s_2
e_7	–	–	A	Login	t_7	s_3
e_8	2	5	–	Pay	t_8	s_3

Definition 8 (Contrast vector) Given an event e', and the sequence of current customer $\mathbb{S}$, we can have a vector $V(e') = \{v_1, ..., v_M\}$, where

$$v_i = 1 - \frac{\left|\left\{e | e \in \mathbb{S}, e.c = e'.c, e.a_i = e'.a_i\right\}\right|}{|\{e | e \in \mathbb{S}, e.c = e'.c\}|} \tag{3.3}$$

Here M is the number of all attributes involved by events, and $a_i (1 \leq i \leq M)$ is the value of attribute A_i. We call $V(e')$ is **contrast vector** of e'. The contrast vector is a set of metrics to evaluate the support for features of the current event among historical activities of the customer.

Before calculating the contrast vector of e', we usually first discretize all the numeric attributes. For some nominal attributes that have too many distinct values, we also categorize them into several groups. In the pattern mining phase, the contrast vector can be served as derived features.

Suppose e_8 in Table 3 is the target event to be predicted. Since e_8 and e_7 occur in the same session s_3 and they are two different event types with different attributes, we merge e_7 and e_8 to calculate the contrast vector. According to the definition of contrast vector, we have 3 elements in $V(e_8)$, where v_1 and v_2 are to evaluate the contrast in attributes A_1 and A_2 respectively. So $v_1 = 1 - 3/4 = 1/4$, and $v_2 = 1 - 1/4 = 3/4$. v_3 is the contrast of login event e_7, so $v_3 = 1 - 2/2 = 0$. Then we get $V(e_8) = \{1/4, 3/4, 0\}$. In the next phase, $V(e_8)$ can be merged with the basic features of e_8 when mining contrast patterns.

The underlying problem in online banking fraud detection is to extract event sequences, form a sequence database, build a contrast vector for each customer by linking to his/her event sequences, and then identify discriminant patterns. Such patterns fire alerts that indicate the risk of leading to online banking fraud. In the following sections, we will introduce the system framework and algorithms to support the detection of online banking fraud.

4 System framework

We have implemented an online banking risk management system: i-Alertor. i-Alertor integrates various features and data mining models, and aims to consolidate different sources of resources for systematic problem-solving.

Figure 1 shows the architecture of our proposed online banking fraud detection system, i-Alertor. i-Alertor consists of four tiers: database, data pre-processing, modeling and alerting, based on the mining process. The database tier locates data resources and connects them to retrieve related data. Relevant online banking fraud detection data is collected from heterogeneous data sources, including Internet banking real time transaction logs, recent and historical transaction data, customer demographic data, and other external sources. The types and format of the data also vary from different sources. To reduce the high volume of source data, we extract relevant information from raw data and transform it into required formats for the models. The relevant data involves realtime online banking transactions, customer banking behavior sequences, historical data, and customer profiles.

The pre-processing tier is in charge of real-time transaction accumulation, historical data maintenance, and preparing the data for model training and prediction. It

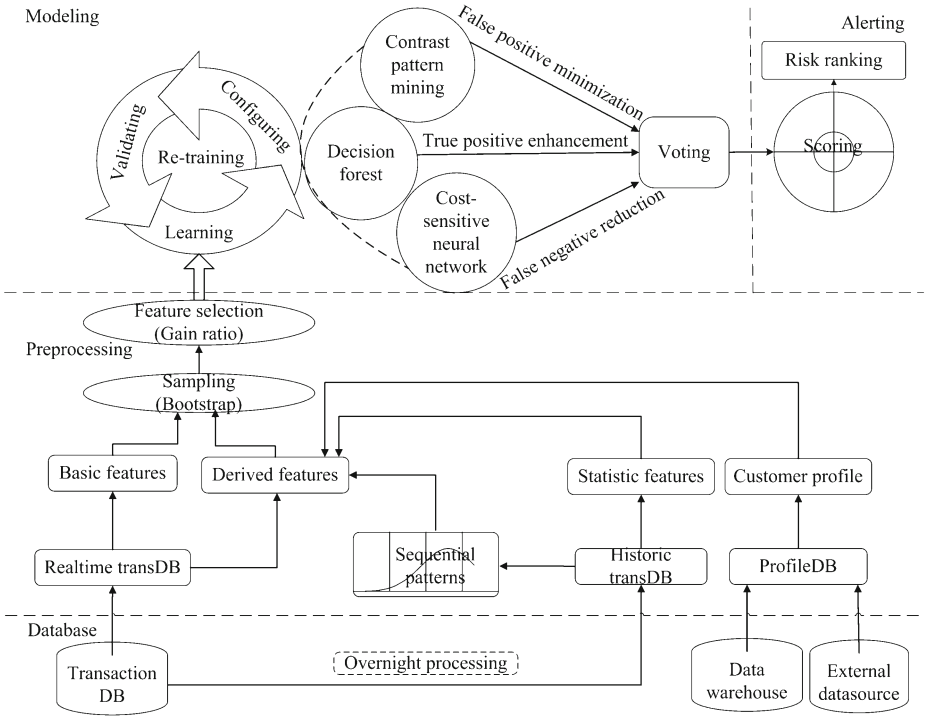

Figure 1 Framework for online banking fraud detection.

also includes the function for selecting basic features and deriving features. There are two main tasks in the data preprocessing stage: data sampling and feature selection. As the online banking transaction data is extremely imbalanced, sampling is necessary before applying any data mining models [16]. Because the number of genuine transactions is huge, we use under sampling to reduce the data volume and class distribution imbalance [27]. In our system, bootstrap sampling [17, 21] is applied to keep the data's statistical distribution. Feature selection [31] is also crucial for the model. Too many features used in the model will highly affect its efficiency, which is very important for real time fraud detection. In our system, we calculate the relevance of a feature to each class in terms of its information gain ratio between two classes, and choose those which have relatively higher information gain ratio. They are the features with the highest discrimination.

The modeling tier provides the generation of models, such as model formation, parameter setting, task scheduling, model retraining, etc. Three data mining methods are adopted in the system:

– contrast pattern mining, which identifies contrast banking behavior highly associated with online banking fraud;
– cost-sensitive neural network, which emphasizes the higher cost of making an error in the misclassification of a fraud compared to a genuine transaction;

– decision forest, which combines the power of individual decision trees in a weighted manner by cascading-and-sharing for constructing decision tree ensembles;

The alerting tier combines outputs from these three models according to a voting method in terms of their scores for each transaction. Finally, a risk score is generated for each transaction. An alert may be fired if a transaction has a score higher than the given threshold. The combination of three methods reduces the false positive rate and false negative rate, and increases the true positive rate.

Below, we first briefly discuss the three models and their combination. Because the classic contrast pattern mining methods are not suitable for online banking fraud detection, a new method called ContrastMiner for contrast online banking behavior mining is presented in Section 5, and the details of the risk scoring based on model combination are given in Section 6.

4.1 Contrast pattern mining

Definition 9 (Contrast Patterns) Given two transaction data sets, D_f and D_g, D_f contains the fraud data samples, and D_g contains the genuine data samples. Let $S_{D_f}(X)$ denote the support of item set X in D_f and $S_{D_g}(X)$ be the support of X in D_g, and then the **Contrast Patterns (CPs)** can be defined as below:

$$CPs = \left\{X | S_{D_g}(X) \leq \omega * S_{D_f}(X), S_{D_f}(X) \geq \theta\right\} \tag{4.1}$$

$\omega > 0$ is the *contrast coefficient* and θ is the threshold of *minimal detection rate*.

Further, we define the following *contrast function* to denote the support difference of an item set X in two data sets D_f and D_g,

$$F(X) = \frac{S_{D_f}(X)}{S_{D_g}(X)} \tag{4.2}$$

For the convenience of explanation, all item sets are projected into a support plane, as shown in Figure 2, where Y-axis stands for the support of an item-set in the genuine data set (D_g), and X-axis is the support in the fraud set (D_f). The contrast patterns are all located in the trapezoid $ABDE$.

The typical method for contrast pattern mining is Emerging Pattern Mining [24]. As a special case of emerging pattern mining, Jumping Emerging Pattern (JEP) mining is proposed to identify strongly contrasting item sets, whose support in D_g is zero but that in D_f equals to or is greater than a threshold specified for D_f. In fact, JEPs rarely exist in class imbalanced data sets, such as the online banking transaction database, where the minority class is set as the target D_f.

As mentioned before, contrast pattern mining is an NP hard problem and its time cost is high especially when applied to data sets with a very small threshold of minimal detection rate. Therefore, Emerging Pattern Mining can not perform well in detecting fraud in online banking. It has also been proved by our experiments.

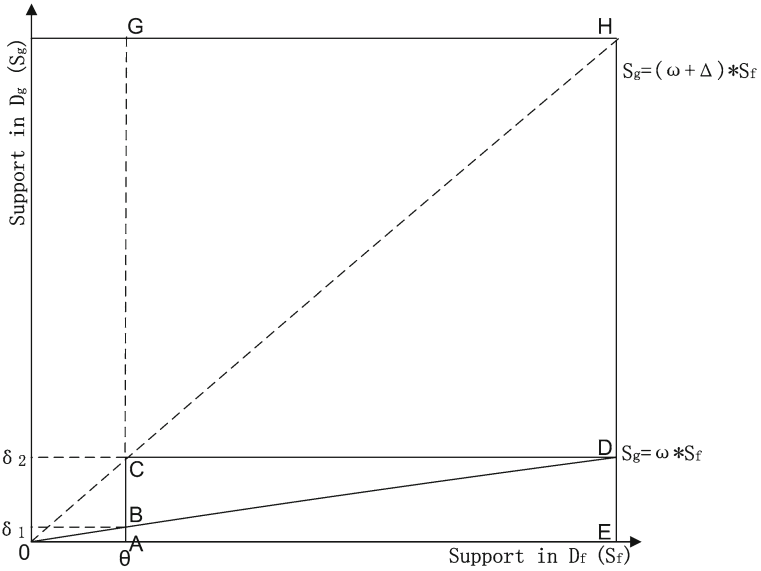

Figure 2 Support-based contrast pattern projection plane.

We observe the following reasons that show its limitations in discovering patterns in extremely imbalanced data:

(1) MDB-LL$_{\text{border}}$ uses the Max-miner algorithm [7] to directly calculate the borders both in D_f and D_g no matter how small the minimum support is. It is usually outside the capability of Max-miner to present the results for extremely small support within an acceptable time period.
(2) MDB-LL$_{\text{border}}$ iterates the whole space of item sets during its border differentiation operation, which is time consuming, especially when the borders contain long patterns.
(3) Emerging pattern mining algorithms usually output a huge number of patterns even with a reasonable growth rate. It is therefore important to have a more effective filter to eliminate insignificant patterns in extremely imbalanced data.

To solve the problems mentioned above, we proposed the ContrastMiner algorithm by improving the MDB-LLborder algorithm through certain strategies, which will be explained in detail in Section 5.

4.2 Cost-sensitive neural network

Cost-sensitive Neural Network (CNN) is a modified neural network-based scoring method, which is especially designed for the online banking scenario.

The extremely imbalanced classification problem in online banking fraud detection makes the classic classification methods (such as statistic method, cost-SVM, decision tree, contrasting pattern based classifier, Bayesian network, neural network) difficult to perform well. However, it was found that neural network still outperformed others in both accuracy and efficiency, and cost-sensitive learning is proven to be a good solution to the class imbalance problem [62]. Therefore, we

extended the idea of neural network and designed a cost-sensitive neural network model. It achieves much better prediction performance compared to the other methods mentioned above.

Artificial neural networks are relatively crude electronic networks of "neurons" based on the neural structure of the brain. They process records one at a time, and "learn" by comparing their classification of the record (which, at the outset, is largely arbitrary) with the known actual classification of the record. The errors from the initial classification of the first record are fed back into the network, and used to modify the networking algorithm in the second round, and so on for many iterations. The error rate is the key parameter for deciding when the iteration can stop. A proper error rate can guarantee good quality of training and high prediction capability.

There are two kinds of error in the training phase:

(1) Positive Error (PE): The PE error determines how different a neural network's output is from the ideal output for genuine transactions, focusing on those genuine transactions mistakenly classified as fraud.
(2) Negative Error (NE): The NE error determines how different a neural network's output is from the ideal output for fraud transactions, focusing on those fraud transactions mistakenly classified as genuine.

For fraud detection, the loss by NE is far more expensive than that of PE. However, the traditional neural network treats cost of PE and NE equally. Therefore, it can gain overall low error even with very high NE in imbalanced data classification, resulting low prediction accuracy of the model. Hence, we modified the neural network model by considering the higher impact of NE in the training phrase, which forms the idea of a cost-sensitive neural network.

4.3 Decision forest

Decision tree is widely used in analysis, and the rules generated by decision tree are easy to understand. However, there are some disadvantages when applying decision tree in fraud detection. The generated rules are biased towards certain features. In a cost-sensitive scenario, there are also too many small branches in the tree and the overfitting problem is serious. Also, since decision tree has only one root, the rules generated are usually locally important but globally ineffective.

To discover more effective rules, we introduce a decision forest, which consists of multiple strong decision trees. It works better than the classic decision tree methods on imbalanced data in building a scoring model.

5 Mining contrast banking behavior

This section introduces the model for identifying contrast patterns in online banking behavior.

5.1 Framework

The framework of contrast behavior mining is shown in Figure 3. Besides the two source data sets: relational transaction database and sequential database, the system

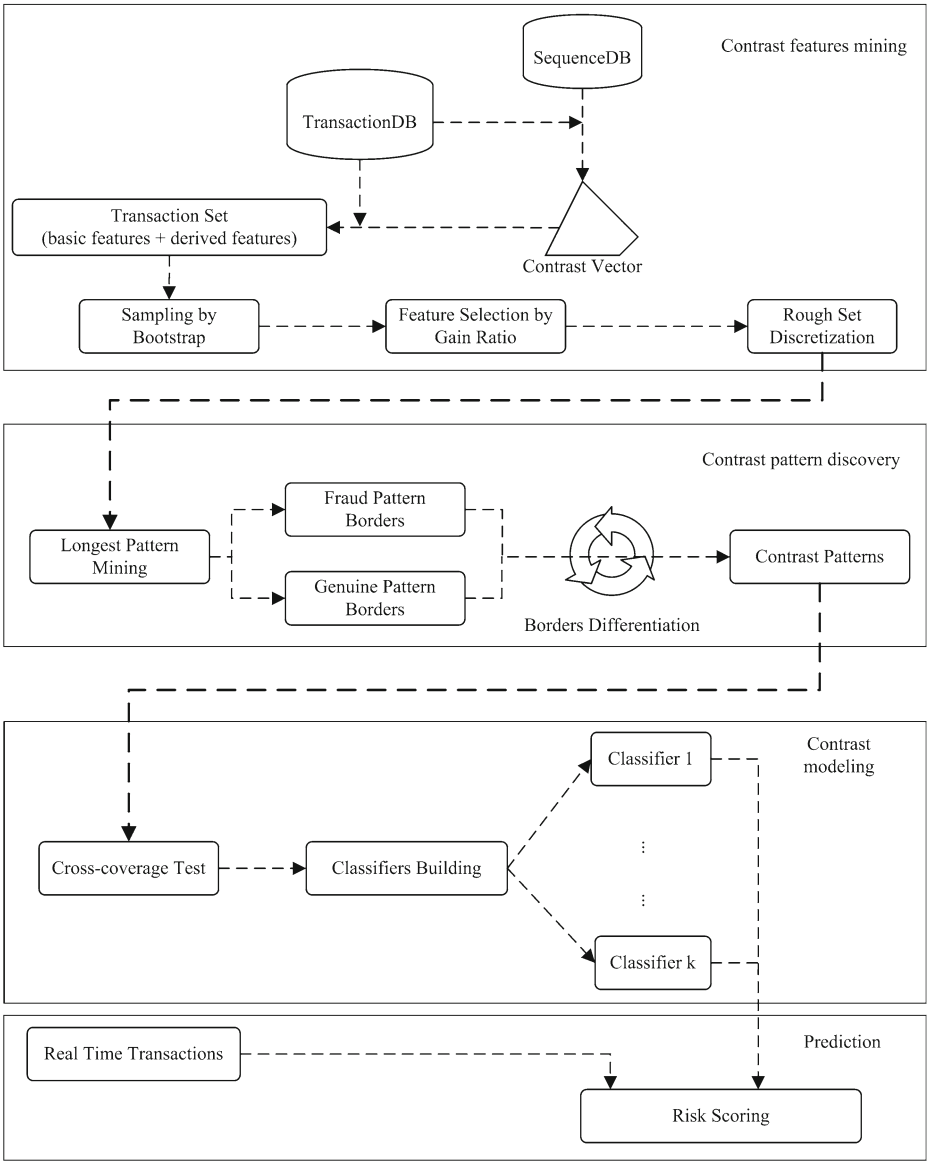

Figure 3 Framework for mining contrast online banking behavior.

consists of four stages: contrast feature mining, contrast pattern discovery, multiple method-based contrast modeling, and risk scoring by contrast. Stage one mainly focuses on data pre-processing. Firstly, the sequence of each customer is constructed from the customer's online banking transactions, which is used to generate the contrast vector for each transaction. We then combine the basic features and contrast vector (derived features) to form the raw feature set. Bootstrap sampling is adopted to perform under-sampling to obtain the training set. We then use the information

gain ratio to select significant features whose weights are higher than the specified threshold. Finally, rough set is introduced to discretize the attributes we have selected.

Stage two is the core step to mine contrast patterns. Initially, Max-miner is used to create pattern borders for both fraud and genuine sets. The border-differentiation and validation process are executed to output the contrast patterns constantly.

Stage three builds the model group powered by the patterns output in the previous stage. A cross-coverage test selects the best union of pattern set, which effectively represents the classification power among all the patterns. The sample sets used in the cross-coverage test will be divided into multiple parts, and each part generates one model. As a result, we obtain multiple models to do the risk scoring.

The last stage is the real time prediction. The scores given by multiple models for one transaction are aggregated by considering each model's weight, which is decided by the coverage rate in the cross-coverage test. A final score is given to a transaction; its value indicates its risk level.

5.2 Modeling sophisticated contrast behavior

We modify MDB-LL$_{\text{border}}$ and improve the efficiency of the Emerging Pattern Mining algorithm [24] by the following strategies:

(1) Instead of calculating pattern borders directly with the small support threshold δ_1 in D_g (see Figure 2), we enhance the minimal support δ_1 to δ_2, which guarantees that Max-miner can succeed in D_g. Then, we use pattern borders in the rectangle ACDE to subtract the borders in the triangle BCD.
(2) In the function Border-Differ [24], iterating the whole sub space of long item sets is time consuming, so all the item sets whose length is bigger than 5 are pushed into a hash table and processed during the validation checking phrase.
(3) We adopt the cross-coverage test, which prunes an extremely high number of redundant patterns, before building our classifiers.

Algorithm 1 introduces the main ideas of the algorithm: ContrastMiner for mining contrast behavior patterns in online banking.

Algorithm 1 ContrastMiner

Data: D_f,D_g,β,θ,len.
Result: Contrast patterns.

1 $S^{'} = \{X|S_{D_f}(X) \geq \theta\}$ /* Retrieve pattern sets in fraud set */
2 $S^{''} = \{X|S_{D_g}(X) \geq \beta\}$ /* Retrieve patterns sets in genuine set */
3 **for** *each* E *in* $S^{'}$ **do**
4 $U = \{u|u \in S^{''}, |E - u| \leq len\}$ /* Execute Border-Differ only for short borders */
5 store BORDER-DIFF$([\emptyset, E], [\emptyset, U])$ in $CandCPs$
6 add $\{u|u \in S^{''}, |E - u| > len\}$ into H /* Store long borders for later checking */
7
8 **for** *each* $[\mathcal{L}, \mathcal{R}]$ *in* $CandCPs$ **do**
9 Check the support of item set in$[\mathcal{L}, \mathcal{R}]$ against H and remove the redundancy;

Steps 1 and 2 calculate the frequent pattern borders for genuine and fraud data sets respectively through Max-miner. Border-Differ is then executed to obtain

candidate patterns, followed by a check of the qualification of each sub-border and the output of contrast patterns.

5.3 Selection of contrast behavior patterns

Though we can control the parameters for contrast pattern mining, many patterns presented are similar in structure. Furthermore, the differentiation capability among patterns varies with their distribution, so we adopt the following method to filter redundancy:

(1) Sort the patterns in contrast descending order, and use strong patterns to prune weak ones. For instance, if pattern p_1 is a sub pattern of p_2 and $F(p_1) \geq F(p_2)$, then p_2 will be removed. The principle of the pruning is that we keep those which are more general, without diminishing the differentiation power.
(2) Conduct the cross-coverage test to choose outstanding patterns. The algorithm for the cross-coverage test is presented in Algorithm 2.

Algorithm 2 Cross-coverage test

```
Data: Pattern set ℙ, coverage threshold η, D_f, D_g
Result: Positive set P^+ and negative set P^−
Initialize matchCount of samples in D_f and D_g to 0
Sort ∀X ∈ ℙ by F(X) descending order and then by |X| ascending order
for each X ∈ ℙ do
    if ∃T ∈ D_f and X ⊆ T then
        P^+ = P^+ ∪ X
        Count(T) = Count(T) + 1
        if Count(T) ≥ η then
            remove T from D_f
    if ∃T ∈ D_g and X ⊆ T then
        P^− = P^− ∪ X
        Count(T) = Count(T) + 1
        if Count(T) ≥ η then
            remove TfromD_g
return P^+ and P^−
```

In the application of anomaly detection, the number of positive samples is usually quite limited compared to that of negative ones. It is difficult and often very costly to acquire a pattern set which can capture all the properties of the negative samples. On the other hand, properties indicating positive behavior are often manipulated and subjected to change, which makes it hard to detect exceptions in imbalanced data. In reality, it is very challenging to select a pattern set that is adaptive enough to capture the dynamics of exceptional behavior. In our method, P^- is selected through the coverage test against D_g, which evaluates how far a transaction is from C_f(fraud label), while P^+ is selected on D_f and provides an indication on how close a transaction belongs to C_f.

6 Risk scoring Internet banking behavior based on combined models

In this section, we discuss the risk scoring model which combines three data mining models for generating the risk rate of each transaction.

6.1 Rationale

Risk scoring evaluates the risk of an individual customer's behavior. After analysis, a score is assigned to each transaction, with a higher score indicating a higher probability of being fraud. Experiments on long term online banking transaction data show that no single existing risk scoring method always works well because of the diversity and dynamics of fraud. We therefore use multiple methods to analyze the risk from different aspects and combine their risk scoring results by a voting strategy. In our system, scores from three models: contrast pattern mining, cost-sensitive neural network and decision forest, are combined by a voting method.

6.2 Risk scoring by individual models

6.2.1 Scoring by contrast pattern mining

The base scores of each pattern X_i in P^+ and P^- are $Base(X_i, P^+)$ and $Base(X_i, P^-)$ respectively, which can be calculated by (6.1) and (6.2):

$$Base\left(X_i, P^+\right) = \left(S_{Df}(X_i) * F(X_i)/(1 + F(X_i))\right) \tag{6.1}$$

$$Base\left(X_i, P^-\right) = \left(S_{Dg}(X_i) * F(X_i)/(1 + F(X_i))\right) \tag{6.2}$$

The scores of each transaction t on D_f and D_g for P^+ and P^- are $\mathbb{S}^+(t)$ and $\mathbb{S}^-(t)$ respectively:

$$\mathbb{S}^+(t) = \sum_{X_i \in P^+, X_i \subseteq t} Base\left(X_i, P^+\right) \Big/ \sum_{X_i \in P^+} Base\left(X_i, P^+\right) \tag{6.3}$$

$$\mathbb{S}^-(t) = \sum_{X_i \in P^-, X_i \subseteq t} Base\left(X_i, P^-\right) \Big/ \sum_{X_i \in P^-} Base\left(X_i, P^-\right) \tag{6.4}$$

Intuitively, suppose there are two transactions t_1 and t_2, if $\mathbb{S}^+(t_1) > \mathbb{S}^+(t_2)$, then the probability for $t_1 \in C_f$ (fraud) is bigger than that of t_2. On the other hand, if $\mathbb{S}^-(t_1) > \mathbb{S}^-(t_2)$ then t_1 is less likely to be classified into C_f than t_2.

Since the scale for $\mathbb{S}^+(t)$ and $\mathbb{S}^-(t)$ is incompatible according to (6.3) and (6.4), it is not easy to combine them directly. In order to merge the cross-fired transactions by $\mathbb{S}^+(t)$ and $\mathbb{S}^-(t)$, we calculate the score rank in P^+ and P^- individually and combine the rank according to (6.5).

The rank of t in D', which is the transaction set to be predicted, is noted as R_t^+ by $\mathbb{S}^+(t)$ in the descending order, and R_t^- by $\mathbb{S}^-(t)$ in the ascending order. The smaller the rank of t is, the higher the probability of being classified in C_f. The overall rank of a transaction t is $\mathbb{S}(t)_{CP}$, which represents the overall risk level of transaction t.

$$\mathbb{S}(t)_{CP} = \lambda_1 * \mathrm{Max}\left(R_t^+, R_t^-\right) + \lambda_2 * \mathrm{Min}\left(R_t^+, R_t^-\right) \tag{6.5}$$

where λ_1 and λ_2 ($\lambda_1 + \lambda_2 = 1$) are the coefficients to control the preference of the decision. In the experiments, we set $\lambda_1 = 0.2$ and $\lambda_2 = 0.8$, which have been proved to achieve overall high accuracy for most of the data sets we tested.

6.2.2 Scoring by cost-sensitive neural network

Each neuron from the output layer of the neural network represents one of the possible classes. An example is classified in the class which corresponds to the neuron with the maximum output. However, the output of the network can also be viewed in the probabilistic sense.

In the online banking fraud detection application, we set the output of the network as the fraud tag, and in the training data set there are only two values for the tag (1 and 0). Therefore, the probability output from the model is used as a risk score for the transaction; The higher the risk score, the more likely that the transaction is a fraud. The transactions are then ranked by this score in descending order. And the rank of the transaction t is noted as $\mathbb{S}(t)_{CNN}$.

6.2.3 Scoring by decision forest

In Algorithm 3, features are first ranked by the gain ratio. The top k features among them are then chosen to build k trees by C4.5 [51], following which the k trees serve as a committee to present the probability of each single transaction being a fraud. The sum of probability will be the risk score output of the model as presented in Algorithm 4. The rank of the transaction t ordered by the risk score output of decision forest descendingly is denoted by $\mathbb{S}(t)_{DF}$.

Algorithm 3 DecisionForest

```
Data: TransactionDB,k
Result: Decision forest
Sort the features by information gain into hash table H;
for each feature F_i in top k of H do
    Build decision tree T_i with F_i as root node;
    Save T_i into tree table TT;
return TT;
```

Algorithm 4 PredictionByDecisionForest

```
Data: TransDB/* transactions for prediction*/,Decision Forest TT
Result: Risk score
for each transaction t in TransDB do
    ScoreSum=0;
    for each tree T_i in TT do
        ScoreSum=ScoreSum+(score of t by given by T_i);
    Output ScoreSum as risk score for t;
return TT;
```

6.3 Risk scoring by combined models

The proper combination of multiple models can effectively leverage the strength of each constituent towards better cumulative performance [13]. In online banking fraud detection, multiple methods may have different scores for the same transaction, and we use a weight vector to aggregate the voting. The final score $\mathbb{S}(t)$ is calculated as follows:

$$\mathbb{S}(t) = w_1 * \mathbb{S}(t)_{CP} + w_2 * \mathbb{S}(t)_{CNN} + w_3 * \mathbb{S}(t)_{DF} \tag{6.6}$$

Here, t is the transaction to be predicted, w_i $(i = 1, 2, 3)$ is the weight of $i - th$ model.

We assigned a weight for each model according to its prediction accuracy on test data. For example, if we adopt two models, Models 1 and 2, and Model 1's weight is 0.8 while Model 2's is 0.2, the aggregated score will be the final score of a transaction.

7 Experiments and evaluation

We built the system i-Alertor as shown in Figure 4. It incorporates the three models (ContrastMiner, CNN and DecisionForest) for online banking fraud detection.

The objectives of the experimental evaluation and the corresponding baseline methods are:

(1) To compare the combined risk scoring model with an existing rule-based system (we call it ExpertSystem) used in the major banks in Australia;
(2) To compare the performance of ContrastMiner with the existing algorithm MDB-LL$_{\text{border}}$ [24] from the perspectives of data scalability, dimension adaptivity and tolerance to imbalance.

7.1 Data

The data set used in our experiment is the online banking transactional data from a major Australian bank. It consists of 8,000,000 genuine transactions (D_g) and 1,500 frauds (D_f), and the number of attributes is 130.

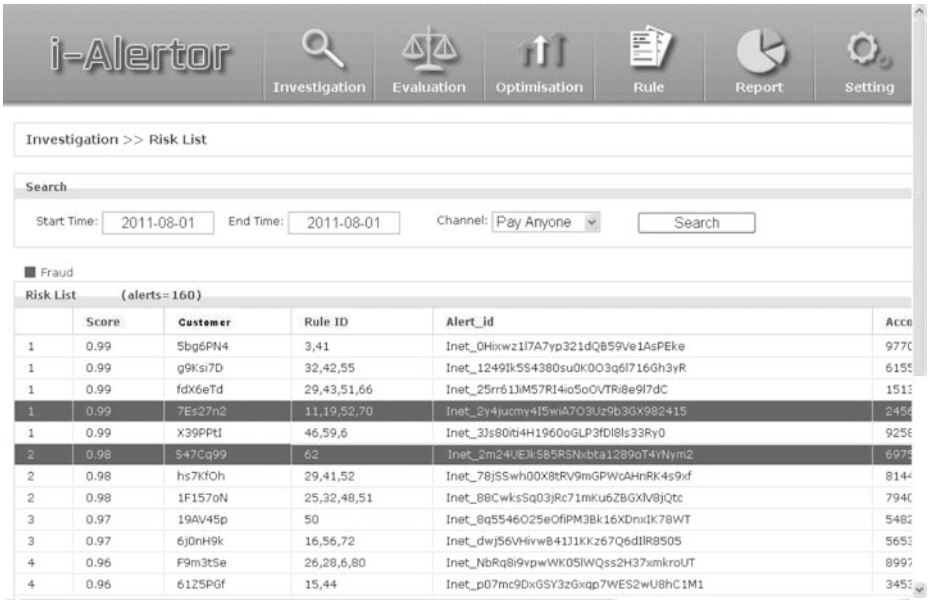

Figure 4 Online banking risk management system: i-Alertor.

7.2 Experimental settings

We evaluated our system i-Alertor against the rule-based online banking fraud detection system ExpertSystem. There are two main metrics to evaluate performance of an online banking fraud detection system.

- *Alert volume*, which is the number of alerts generated every day. According to the business process in the bank, every triggered alert has to be investigated manually for further process. A large number of alerts will make the investigation quite labor-intensive and time-consuming, resulting in high cost.
- *Detection rate*, which is the percentage of fraud detected by the system. The perfect case is one in which the system can catch all fraud, meaning that the detection rate is 100 %.

ContrastMiner and MDB-LL_{border} are both contrast pattern mining algorithms, so their accuracy is the same when applied to the same data set. In order to evaluate the efficiency of ContrastMiner against MDB-LL_{border}, we compare the computational time spent by ContrastMiner and MDB-LL_{border}. For both algorithms, we choose the same level of contrasting rate and compare their performance.

7.3 Overall performance evaluation

Figure 5 is an overview of distribution of fraud caught by ExpertSystem and our system i-Alertor. Here i-Alertor alerted the top 200 riskiest transactions of each day. The figure shows that there were 49.2 % of fraud detected by both systems, while our system can detect an additional 16.8 % of fraud. In total, our system has around 7 % higher detection rate.

Figure 6 shows the detection rate under different alert volumes generated by ExpertSystem, ExpertSystem with ranked alerts, and i-Alertor. As there is no rank among original alerts from ExpertSystem, to obtain top n alerts, the alerts were randomly selected from ExpertSystem's daily alerts. For ranked alerts from ExpertSystem, the alerts are selected according to their risk score rank calculated by

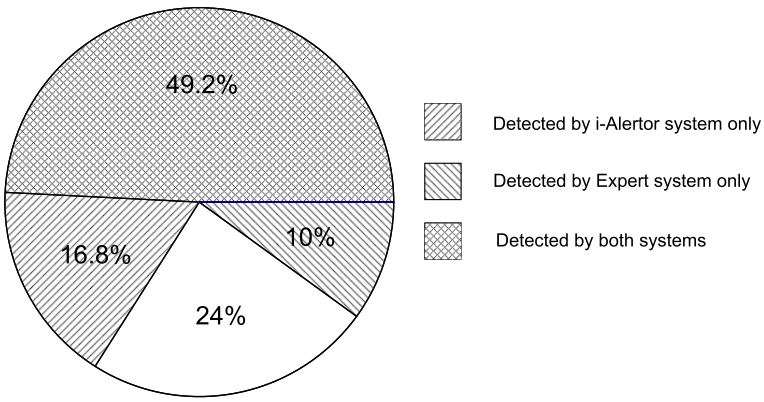

Figure 5 Distribution of fraud detected by different systems.

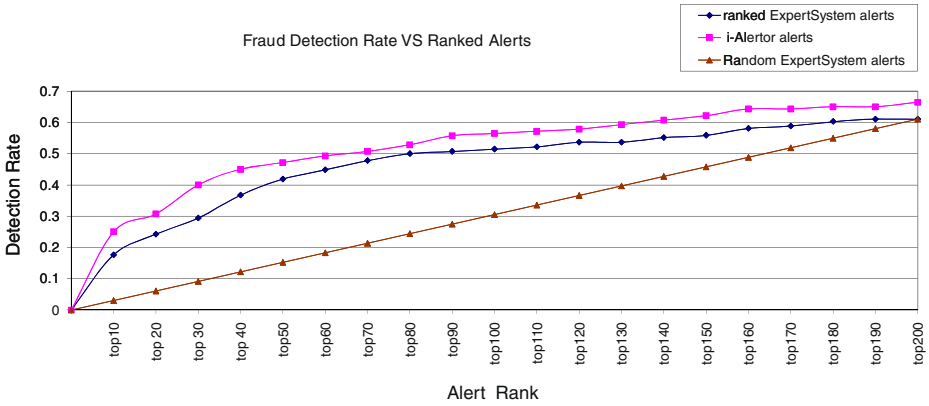

Figure 6 Detection rate comparison between i-Alertor and ExpertSystem with ranked alerts.

i-Alertor. From Figure 6, we can see that ExpertSystem with ranked alerts performed much better than original alerts from ExpertSystem. i-Alertor worked even better than ExpertSystem with ranked alerts; at the same alert volume, i-Alertor sometimes even has a 10 % higher detection rate. Overall i-Alertor always has better performance than ExperSystem. It is mainly because of fraud dynamics and i-Alertor can catch high percentage of new fraud which are missed by rules in the ExpertSystem.

Figure 7 is the evaluation of i-Alertor on missing fraud detection. It shows that i-Alertor can catch 25 % of fraud missed by ExpertSystem within 60 alerts. Therefore, if we combine ExpertSystem with i-Alertor, i-Alertor can help ExpertSystem detect more missing fraud by increasing very small alert volume.

7.4 Performance of contrast behavior modeling

Finally, we assess the performance of ContrastMiner in mining contrast behavior in imbalanced data.

For the efficiency against MDB-LL_{border}, Figure 8 shows that, with the increase of contrasting rate, ContrastMiner highly outperforms MDB-LL_{border}. Because it is

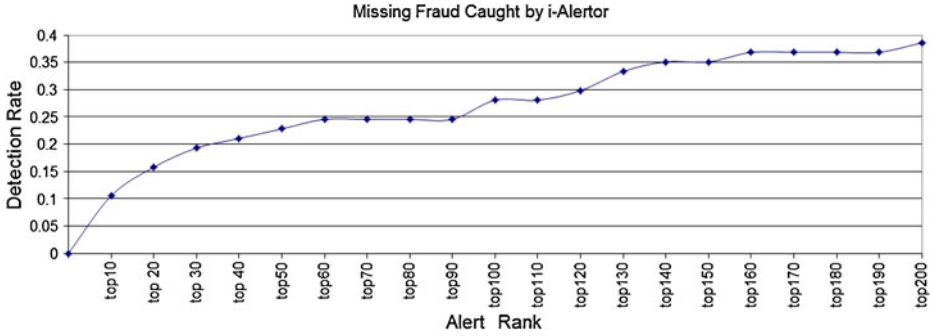

Figure 7 Missing fraud detection rate of i-Alertor.

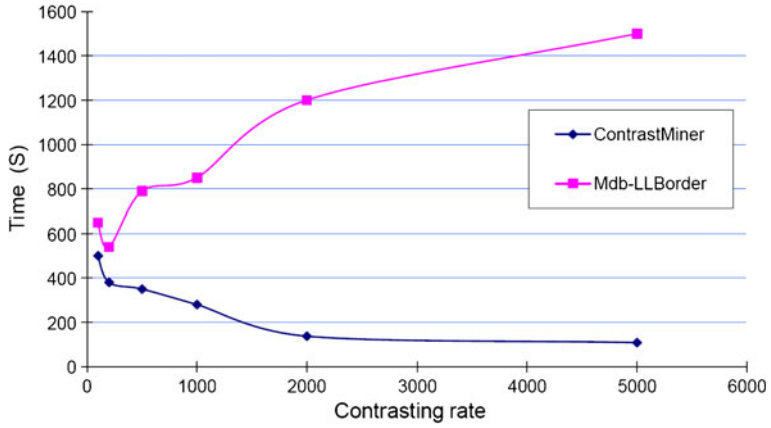

Figure 8 Efficiency of ContrastMiner VS MDB-LL_{border}.

time consuming for MDB-LL_{border} to mine pattern borders especially when support threshold for D_g is extremely small in highly imbalanced scenario. For example, when contrasting rate is set to 2000, which means MDB-LL_{border} needs to discovery all patterns whose support <0.0005, it will be difficult for MDB-LL_{border} to succeed in acceptable time cost. However, ContrastMiner can still work well because it is not impacted by the minimal support. As mentioned in Section 5.2, a small support can be enhanced to a practical value on which Max-miner can succeed easily. So, with the increase of contrast rate, the support to be processed in D_g decreased rapidly, and therefore MDB-LL_{border} costs lots of CPU time to output candidate pattern borders. On the other hand, ContrastMiner takes less time with higher contrast rate, because there is less candidate patterns for selection in Trapezoidal $ABDE$ area after candidate pattern borders are got.

8 Conclusions

Sophisticated online banking fraud involves multiple resources, including human wisdom, computing tools, web technology and online business systems. The instant and effective detection of such fraud challenges existing fraud detection techniques and systems. In this paper, we report our study and practices in the real world. A systematic online banking fraud detection approach is introduced. Its framework takes advantage of domain knowledge, mixed features, multiple data mining methods, and a multiple-layer structure for a systematic solution. It includes three algorithms: contrast pattern mining, neural network and decision forest, and their outcomes are integrated with an overall score measuring the risk of an online transaction being fraudulent or genuine. The approach is particularly effective in detecting fraud in a large volume of extremely imbalanced data. We test the approach and the system in a major bank. Massive experiments show that our framework significantly improves fraud detection accuracy and performs better than existing fraud detection methods and systems in both efficiency and accuracy. The approach can also be combined with the existing banking fraud detection system.

Acknowledgements This work is partially sponsored by the Australian Research Council Discovery grant (DP1096218) and Linkage grant (LP100200774).

References

1. Aggelis, V.: Offline Internet banking fraud detection. In: Proc. of the 1st International Conference on Availability, Reliability and Security, pp. 904–905. IEEE (2006)
2. Aleskerov, E., Freisleben, B., Rao, B.: CARDWATCH: a neural network based database mining system for credit card fraud detection. In: Proc. of Computational Intelligence for Financial Engineering (CIFEr), pp. 220–226. New York, USA (1997)
3. Alfuraih, S.I., Sui, N.T., McLeod, D.: Using trusted email to prevent credit card frauds in multimedia products. World Wide Web **5**(3), 245–256 (2002)
4. Altman, E.I., Marco, G., Varetto, F.: Corporate distress diagnosis: comparisons using linear discriminant analysis and neural networks (the Italian experience). J. Bank. Finance **18**(3), 505–529 (1994)
5. AV-Test.org. http://www.av-test.org/en/statistics/malware/. Accessed 5 Jan 2012
6. Bay, S.D., Pazzani, M.J.: Detecting group differences: mining contrast sets. Data Mining and Knowledge Discovery **5**(3), 213–246 (2001)
7. Bayardo, Jr., R.J.: Efficiently mining long patterns from databases. In: Proc. of the 1998 ACM SIGMOD International Conference on Management of Data, pp. 85–93. New York, USA (1998)
8. Bignell, K.B.: Authentication in an Internet banking environment: towards developing a strategy for fraud detection. In: Proc. of International Conference on Internet Surveillance and Protection (ICISP), Cote d'Azur, France, pp. 23–30. IEEE (2006)
9. Brause, R., Langsdorf, T., Hepp, M.: Neural data mining for credit card fraud detection. In: Proc. of the 11th IEEE International Conference on Tools with Artificial Intelligence, Chicago, USA, pp. 103–106 (1999)
10. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
11. Cao, L., Dai, R.: Open Complex Intelligent Systems. Post & Telecom (2008)
12. Cao, L., Dai, R., Zhou, M.: Metasynthesis: M-Space, M-Interaction and M-Computing for open complex giant systems. IEEE Trans. Syst. Man Cybern., Part A **39**(5), 1007–1021 (2009)
13. Cao, L., Zhang, H., Zhao, Y., Luo, D., Zhang, C.: Combined mining: discovering informative knowledge in complex data. IEEE Trans. Syst. Man Cybern., Part B **41**(3), 699–712 (2011)
14. Chang, R.I., Lai, L.B., Su, W.D., Wang, J.C., Kouh, J.S.: Intrusion detection by backpropagation neural networks with sample-query and attribute-query. Int. J. Comput. Intell. Res. **3**(1), 6–10 (2007)
15. Chanson, S.T., Cheung, T.W.: Design and implementation of a PKI-based end-to-end secure infrastructure for mobile e-commerce. World Wide Web **4**(4), 235–253 (2001)
16. Chawla, N.V.: Data mining for imbalanced datasets: an overview. In: Data Mining and Knowledge Discovery Handbook, pp. 875–886 (2010)
17. Chernick, M.R.: Bootstrap Methods: A Practitioner's Guide, 2nd edn. Wiley Series in Probability and Statistics (2007)
18. Cox, K.C., Eick, S.G., Wills, G.J., Brachman, R.J.: Brief application description; visual data mining: recognizing telephone calling fraud. Data Mining and Knowledge Discovery **1**(2), 225–231 (1997)
19. CyberSource Company: Credit card fraud management. http://www.cybersource.com. Accessed 5 Jan 2012
20. Dandash, O., Wang, Y., Leand, P.D., Srinivasan, B.: Fraudulent Internet banking payments prevention using dynamic key. J. Networks **3**(1), 25–34 (2008)
21. Davison, A.C., Hinkley, D.V.: Bootstrap Methods and Their Application. Cambridge University Press, Cambridge (1997)
22. Deshmukh, A., Talluru, L.: A rule-based fuzzy reasoning system for assessing the risk of management fraud. Int. J. Intell. Syst. Account. Finance Manage. **7**(4), 223–241 (1998)
23. Dheepa, V., Dhanapal, R.: Analysis of credit card fraud detection methods. Int. J. Recent Trends Eng. **2**(3), 126–128 (2009)
24. Dong, G., Li, J.: Efficient mining of emerging patterns: discovering trends and differences. In: Proc. of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, USA, pp. 43–52 (1999)

25. Dong, G., Zhang, X., Wong, L., Li, J.: CAEP: classification by aggregating emerging patterns. In: Proc. of the 2nd International Conference on Discovery Science, Tokyo, Japan, pp.30–42. Springer (1999)
26. Dorronsoro, J.R., Ginel, F., Sgnchez, C., Cruz, C.: Neural fraud detection in credit card operations. IEEE Trans. Neural Netw. **8**(4), 827–834 (1997)
27. Drummond, C., Holte, R.C.: C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In: Workshop on Learning from Imbalanced Datasets II, International Conference on Machine Learning, Washington DC (2003)
28. Edge, K., Raines, R., Grimaila, M., Baldwin, R., Bennington, R., Reuter, C.: The use of attack and protection trees to analyze security for an online banking system. In: Proc. of the 40th Annual Hawaii International Conference on System Sciences (HICSS), Waikoloa, Hawaii (2007)
29. Fan, W., Miller, M., Stolfo, S., Lee, W., Chan, P.: Using artificial anomalies to detect unknown and known network intrusions. Knowl. Inf. Syst. **6**(5), 507–527 (2004)
30. Ghosh, A.K., Schwartzbard, A.: A study in using neural networks for anomaly and misuse detection. In: Proc. of the 8th Conference on USENIX Security Symposium, p. 12. USENIX Association, pp. 141–152. CA, USA (1999)
31. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. J. Mach. Learn. Res. **3**, 1157–1182 (2003)
32. Hassibi, K.: Detecting payment card fraud with neural networks. In: Business Applications of Neural Networks, pp. 141–157 (2000)
33. Hertzum, M., Jrgensen, N., Nrgaard, M.: Usable security and e-banking: ease of use vis-a-vis security. Aust. J. Inf. Syst. **11**(2), 52–65 (2004)
34. Ilgun, K., Kemmerer, R.A., Porras, P.A.: State transition analysis: a rule-based intrusion detection approach. IEEE Trans. Softw. Eng. **21**(3), 181–199 (1995)
35. Karlsen, K.N., Killingberg, T.: Profile based intrusion detection for Internet banking systems. Norwegian University of Science and Technology (2008)
36. Kou, Y., Lu, C.T., Sirwongwattana, S., Huang, Y.P.: Survey of fraud detection techniques. In: Proc. of International Conference on Networking, Sensing and Control, pp. 749–754. IEEE (2004)
37. Kovach, S., Ruggiero, W.V.: Online banking fraud detection based on local and global behavior. In: Proc. of the Fifth International Conference on Digital Society, Guadeloupe, France, pp. 166–171 (2011)
38. Kumar, S., Spafford, E.H.: A pattern matching model for misuse intrusion detection. In: Proc. of the National Computer Security Conference, pp. 11–21 (1994)
39. Leung, A., Yan, Z., Fong, S.: On designing a flexible e-payment system with fraud detection capability. In: Proc. of IEEE International Conference on e-Commerce Technology, pp. 236–243. IEEE (2004)
40. Lee, W., Stolfo, S.J.: Data mining approaches for intrusion detection. In: Proc. of the 7th Conference on USENIX Security Symposium. Usenix Association, CA, USA (1998)
41. Li, J., Dong, G., Ramamohanarao, K.: Making use of the most expressive jumping emerging patterns for classification. Knowl. Inf. Syst. **3**(2), 131–145 (2001)
42. Maes, S., Tuyls, K., Vanschoenwinkel, B., Manderick, B.: Credit card fraud detection using Bayesian and neural networks. In: Interactive Image-Guided Neurosurgery, pp. 261–270 (1993)
43. Mahdi, M.D.H., Rezaul, K.M., Rahman, M.A.: Credit fraud detection in the banking sector in UK: a focus on e-business. In: Proc. of the 4th International Conference on Digital Society (ICDS '10), St. Maarten, pp. 232–237 (2010)
44. Mannan, M., van Oorschot, P.C.: Security and usability: the gap in real-world online banking. In: Proc. of the 2007 Workshop on New Security Paradigms (NSPW '07), pp. 1–14. NY, USA (2008)
45. Moreau, Y., Preneel, B., Burge, P., Shawe-taylor, J., Stoermann, C., Ag, S., Vodafone, C.C.: Novel techniques for fraud detection in mobile telecommunication networks. In: Proc. of ACTS Mobile Summit, Granada, Spain (1997)
46. Neill, D.B., Moore, A.W.: Rapid detection of significant spatial clusters. In: Proc. of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 256–265. NY, USA (2004)
47. Papazoglou, M.P.: Web services and business transactions. World Wide Web **6**(1), 49–91 (2003)
48. Phua, C., Alahakoon, D., Lee, V.: Minority report in fraud detection: classification of skewed data. ACM SIGKDD Explor. Newsl. **6**(1), 50–59 (2004)
49. Phua, C., Lee, V., Smith, K., Gayler, R.: A comprehensive survey of data mining-based fraud detection research. Arxiv preprint arXiv:1009.6119 (2010). Accessed 5 Jan 2012

50. Quah, J.T.S., Sriganesh, M.: Real-time credit card fraud detection using computational intelligence. Expert Syst. Appl. **35**(4), 1721–1732 (2008)
51. Quinlan, J.R.: C4. 5: Programs for Machine Learning. Morgan Kaufmann (1993)
52. Ramamohanarao, K., Fan, H.: Patterns based classifiers. World Wide Web **10**(1), 71–83 (2007)
53. Rosset, S., Murad, U., Neumann, E., Idan, Y., Pinkas, G.: Discovery of fraud rules for telecommunications challenges and solutions. In: Proc. of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 409–413. NY, USA (1999)
54. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach, 3rd edn. Prentice Hall (2010)
55. Ryan, J., Lin, M.J., Miikkulainen, R.: Intrusion detection with neural networks. In: Proc. of Conference on Advances in Neural Information Processing Systems, pp. 943–949. MIT Press (1997)
56. Srivastava, A., Kundu, A., Sural, S., Majumdar, A.K.: Credit card fraud detection using hidden Markov model. IEEE Trans. Dependable Secure Comput. **5**(1), 37–48 (2008)
57. Syeda, M., Zhang, Y.Q., Pan, Y.: Parallel granular neural networks for fast credit card fraud detection. In: Proc. of International Conference on Fuzzy Systems, HI, USA, pp. 572–577 (2002)
58. Smaha, S., Winslow, J.: Misuse detection tools. Comput. Secur. J. **10**(1), 39–49 (1994)
59. Taniguchi, M., Haft, M., Hollmén, J., Tresp, V.: Fraud detection in communication networks using neural and probabilistic methods. In: Proc. of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, WA, USA, pp. 1241–1244 (1998)
60. Ureche, O., Plamondon, R.: Digital payment systems for Internet commerce: the state of the art. World Wide Web **3**(1), 1–11 (2000)
61. Wang, L., Zhao, H., Dong, G., Li, J.: On the complexity of finding emerging patterns. Theor. Comp. Sci. **335**(1), 15–27 (2005)
62. Weiss, G.M.: Mining with rarity: a unifying framework. ACM SIGKDD Explor. Newsl. **6**(1), 7–19 (2004)
63. WI-IAT 2011 Panel on Wisdom Web of Things (W2T): Fundamental issues, challenges and potential applications. wi-iat2011.org. Accessed 5 Jan 2012
64. Zhong, N., Liu, J., Yao, Y.Y.: In search of the wisdom web. IEEE Comput. **35**(11), 27–31 (2002)
65. Zhong, N., Liu, J., Yao, Y.Y.: Envisioning intelligent information technologies through the prism of web intelligence. Commun. ACM **50**(3), 89–94 (2007)
66. Zhong, N., Ma, J.H., Huang, R.H., Liu, J.M., Yao, Y.Y., Zhang, Y.X., Chen, J.H.: Research challenges and perspectives on Wisdom Web of Things (W2T). J. Supercomputing (2010). doi:10.1007/s11227-010-0518-8
67. Zhou, Z.H., Liu, X.Y.: Training cost-sensitive neural networks with methods addressing the class imbalance problem. IEEE Trans. Knowl. Data Eng. **18**(1), 63–77 (2006)