Debt Detection in Social Security by Sequence Classification Using Both Positive and Negative Patterns

Yanchang Zhao[1], Huaifeng Zhang[2], Shanshan Wu[1], Jian Pei[3], Longbing Cao[1], Chengqi Zhang[1], and Hans Bohlscheid[1,2]

[1] Centre for Quantum Computation and Intelligent Systems, Faculty of Engineering & IT, University of Technology, Sydney, Australia
{yczhao,shanshan,lbcao,chengqi}@it.uts.edu.au
[2] Business Integrity Review Operations Branch, Centrelink, Australia
{huaifeng.zhang,hans.bohlscheid}@centrelink.gov.au
[3] School of Computing Science, Simon Fraser University, Canada
jpei@cs.sfu.ca

Abstract. Debt detection is important for improving payment accuracy in social security. Since debt detection from customer transactional data can be generally modelled as a fraud detection problem, a straightforward solution is to extract features from transaction sequences and build a sequence classifier for debts. The existing sequence classification methods based on sequential patterns consider only positive patterns. However, according to our experience in a large social security application, negative patterns are very useful in accurate debt detection. In this paper, we present a successful case study of debt detection in a large social security application. The central technique is building sequence classification using both positive and negative sequential patterns.

Keywords: sequence classification, negative sequential patterns.

1 Introduction and Application Background

Centrelink Australia (http://www.centrelink.gov.au) is a Commonwealth Government agency delivering a wide range of services to the Australian community. It is one of the largest data intensive applications in Australia. For example, in financial year 2004-2005 (from 1 July 2004 to 30 June 2005), Centrelink distributes approximately 63 billion dollars in social security payments to 6.4 million customers, makes 9.98 million individual entitlement payments, and records 5.2 billion electronic customer transactions [5].

Qualification for payment of an entitlement is assessed against a customer's personal circumstance. If all criteria are met, the payment to a customer continues until a change of the customer's circumstance precludes the customer from obtaining further benefit. However, for various reasons, customers on benefit payments or allowances sometimes get overpaid. The overpayments collectively lead to a large amount of debt owed to Centrelink. For instance, in financial year

W. Buntine et al. (Eds.): ECML PKDD 2009, Part II, LNAI 5782, pp. 648–663, 2009.

2004-2005, Centrelink raised over $900 million worth of customer debts (excluding Child Care Benefits and Family Tax Benefits) [5]. To achieve high payment accuracy, detection of debts is one of the most important tasks in Centrelink. Centrelink uses a number of processes to determine customers at risk of incurring a debt, however, the processes used only find a certain types of debts, for example, earnings-related debts. Another problem is that processes are applied to all customers, so a lot of time and efforts are spent on customers who are subsequently identified as being non-debtors. In this paper, we discuss our case study of debt detection in Centrelink using data mining techniques.

Debt detection can be generally modelled as a fraud detection problem. Therefore, we can adopt a classification approach. All transactions about a customer form a transaction sequence. If no debt happens to a customer, the sequence is labelled as normal (i.e., no-debt). If a debt happens to a customer, the corresponding customer sequence is labelled as debt. We can collect a training set containing both no-debt and debt sequences and learn a sequence classifier. The classifier can then be applied to new customer sequences to detect possible debts.

A classifier needs to extract features for classification. Since sequences are the data objects in debt detection, it is natural to use sequential patterns, i.e., subsequences that are frequent in customer sequences, as features. The traditional techniques for sequential pattern based classifiers consider only positive patterns, which capture a set of positively correlated events. Moreover, to detect debt at an early stage and prevent debt occurrence, a classification model is needed to predict the likelihood of debt occurrence based on the transactional activity data. Nevertheless, to the best of our knowledge, there are no techniques for building classifiers based on negative sequential patterns like $A \rightarrow \neg B$, $\neg A \rightarrow B$ and $\neg A \rightarrow \neg B$, where A and B and sequential patterns.

To tackle the above problems, based on our previous work on *negative sequential patterns* [27,28], we designed a new technique, *sequence classification using both positive and negative patterns*, to build sequence classifiers with relationship between activity sequences and debt occurrences. The contributions of this paper are:

- A new technique of *sequence classification using both positive and negative sequential patterns*; and
- An application in social security, demonstrating: 1) the effectiveness of our previous technique on *negative sequential pattern mining* to find both positive and negative sequential patterns; and 2) the effectiveness of our new technique on sequence classification using both positive and negative sequential patterns.

The rest of this paper is organized as follows. Our proposed technique of sequence classifiers using both positive and negative sequential patterns is described in Section 2. An application of the above technique in social security is presented in Section 3. Section 4 presents the related work on negative sequential pattern mining, sequence classification and existing systems and applications for fraud detection. Section 5 concludes this paper.

2 Sequence Classification Using Both Positive and Negative Sequential Patterns

From the data mining perspective, sequence classification is to build classifiers using sequential patterns. To the best of our knowledge, all of the existing sequence classification algorithms use positive sequential patterns only. However, the sequential patterns negatively correlated to debt occurrence are very important in debt detection. In this section, we first introduce negative sequential patterns and then propose a novel technique for sequence classification using both negative and positive sequential patterns.

2.1 Negative Sequential Patterns

Traditional sequential pattern mining deals with positive correlation between sequential patterns only, without considering negative relationship between them. To find negative relationship in sequences, we previously designed a notion of *negative sequential rules* [27,28] as follows.

Definition 1. *A negative sequential rule (NSR) is in the form of* $A \rightarrow \neg B$, $\neg A \rightarrow B$ *or* $\neg A \rightarrow \neg B$, *where* A *and* B *are sequential patterns.*

Based on the above definition, there are four types of sequential rules, including the tradition positive sequential rules (see Type I).

- Type I: $A \rightarrow B$, which means that pattern A is followed by pattern B;
- Type II: $A \rightarrow \neg B$, which means that pattern A is not followed by pattern B;
- Type III: $\neg A \rightarrow B$, which means that if pattern A does not appear, then pattern B will occur; and
- Type IV: $\neg A \rightarrow \neg B$, which means that if pattern A does not appear, then pattern B will not occur.

For types III and IV whose left sides are the negation of a sequence, there is no time order between the left side and the right side. Note that A and B themselves are sequential patterns, which make them different from negative association rules. The supports, confidences and lifts of the above four types of sequential

Table 1. Supports, Confidences and Lifts of Four Types of Sequential Rules

	Rules	Support	Confidence	Lift
I	$A \rightarrow B$	$P(AB)$	$\frac{P(AB)}{P(A)}$	$\frac{P(AB)}{P(A)P(B)}$
II	$A \rightarrow \neg B$	$P(A)-P(AB)$	$\frac{P(A)-P(AB)}{P(A)}$	$\frac{P(A)-P(AB)}{P(A)(1-P(B))}$
III	$\neg A \rightarrow B$	$P(B)-P(A\&B)$	$\frac{P(B)-P(A\&B)}{1-P(A)}$	$\frac{P(B)-P(A\&B)}{P(B)(1-P(A))}$
IV	$\neg A \rightarrow \neg B$	$1-P(A)-P(B)+P(A\&B)$	$\frac{1-P(A)-P(B)+P(A\&B)}{1-P(A)}$	$\frac{1-P(A)-P(B)+P(A\&B)}{(1-P(A))(1-P(B))}$

rules are shown in Table 1. In the table, $P(A\&B)$ denotes the probability of the concurrence of A and B in a sequence, no matter which one occurs first, or whether they are interwoven with each other.

2.2 Sequence Classification

Let $\mathcal{S}$ be a sequence dataset and $\mathcal{T}$ be a finite set of *class labels*. A *sequence classifier* is a function

$$\mathcal{F} : \mathcal{S} \to \mathcal{T}. \tag{1}$$

In sequence classification, a classifier $\mathcal{F}$ is built with frequent *classifiable sequential patterns* $\mathcal{P}$.

Definition 2. *A Classifiable Sequential Pattern (CSP) is in the form of* $p_a \to \tau$, *where* τ *is a class ID and* p_a *is a frequent pattern in the sequence dataset* $\mathcal{S}$.

The *support* of a sequential pattern p_a is the proportion of sequences containing p_a, and a sequential pattern is *frequent* in a dataset if its support in the dataset exceeds a user-specified minimum support threshold. Based on the mined sequential patterns, a sequence classifier can be formulized as

$$\mathcal{F} : \mathcal{S} \xrightarrow{\mathcal{P}} \mathcal{T}, \tag{2}$$

where $\mathcal{P}$ is a set of classifiable sequential patterns. That is, for each sequence $s \in \mathcal{S}$, $\mathcal{F}$ predicts the target class label of s based on the sequence classifier built using the classifiable sequential pattern set $\mathcal{P}$. A sequence instance s is said to be *covered* by a classifiable sequential pattern p ($p \in \mathcal{P}$) if s contains p_a, the antecedent of p.

2.3 Discriminative Sequential Patterns

Given a sequence dataset $\mathcal{S}$ and a set of target classes $\mathcal{T}$, a number of frequent classifiable sequential patterns need to be discovered for building a sequence classifier. The conventional algorithms use only positive sequential patterns to build classifiers. However, negative sequential patterns can also contribute to classification. To achieve better classification results, we use both negative and positive sequential patterns to build classifiers. Furthermore, instead of using the complete set of frequent patterns, we select a small set of discriminative classifiable sequential patterns according to Class Correlation Ratio (CCR) [22].

CCR measures how much a sequential pattern p_a is correlated with the target class τ compared to the negative class $\neg\tau$. Based on the contingency table (see Table 2), CCR is defined as

$$CCR(p_a \to \tau) = \frac{\hat{corr}(p_a \to \tau)}{\hat{corr}(p_a \to \neg\tau)} = \frac{a \cdot (c+d)}{c \cdot (a+b)}, \tag{3}$$

Table 2. Feature-Class Contingency Table

	p_a	$\neg p_a$	$\sum$
τ	a	b	$a+b$
$\neg\tau$	c	d	$c+d$
$\sum$	$a+c$	$b+d$	$n=a+b+c+d$

where $\hat{corr}(p_a \rightarrow \tau)$ is the correlation between p_a and the target class τ, defined as

$$\hat{corr}(p_a \rightarrow \tau) = \frac{sup(p_a \cup \tau)}{sup(p_a) \cdot sup(\tau)} = \frac{a \cdot n}{(a+c) \cdot (a+b)}. \tag{4}$$

CCR falls in $[0, +\infty)$. $CCR = 1$ means that the antecedent is independent of the target class. $CCR < 1$ indicates that the antecedent is negatively correlated with the target class, while $CCR > 1$ suggests a positive correlation between them.

In order to use the mined classifiable sequential patterns to build a classifier, we need to rank the patterns according to their capability to make correct classification. The ranking is based on a weighted score

$$W_s = \begin{cases} CCR, & \text{if } CCR \geq 1 \\ \frac{1}{CCR}, & \text{if } 0 < CCR < 1 \\ M, & \text{if } CCR = 0 \end{cases}, \tag{5}$$

where M is the maximum W_s of all rules where $CCR \neq 0$.

2.4 Building Sequence Classifiers

Our algorithm for building a sequence classifier with both positive and negative sequential patterns is composed of five steps.

1) Finding negative and positive sequential patterns using a negative sequential pattern mining algorithm, such as our previous techniques [27,28].
2) Calculating the frequency, chi-square and CCR of every classifiable sequential pattern, and only those patterns meeting *support*, *significance* (measured by chi-square) and *CCR* criteria are extracted into the classifiable sequential pattern set $\mathcal{P}$.
3) Pruning patterns in the obtained classifiable sequential pattern set with the pattern pruning algorithm in [13]. The only difference is that, in our algorithm, *CCR*, instead of confidence, is used as the measure for pruning.
4) Conducting serial coverage test by following the ideas in [15,13]. The patterns which can correctly cover one or more training samples in the test are kept for building a sequence classifier.
5) Ranking selected patterns with W_s and building the classifier as follows. Given a sequence instance s, all the classifiable sequential patterns covering s are extracted. The sum of the weighted score corresponding to each target class is computed and then s is assigned with the class label corresponding to the largest sum.

Table 3. Examples of Activity Transaction Data

Person_ID	Activity_Code	Activity_Date	Activity_Time
******002	DOC	20/08/2007	14:24:13
******002	RPT	20/08/2007	14:33:55
******002	DOC	05/09/2007	10:13:47
******002	ADD	06/09/2007	13:57:44
******002	RPR	12/09/2007	13:08:27
******002	ADV	17/09/2007	10:10:28
******002	REA	09/10/2007	07:38:48
******002	DOC	11/10/2007	08:34:36
******002	RCV	11/10/2007	09:44:39
******002	FRV	11/10/2007	10:18:46
******002	AAI	07/02/2008	15:11:54

3 A Case Study

Our technique was applied in social security to study the relationship between transactional activity patterns and debt occurrences and build sequence classifiers for debt detection.

3.1 Data

The data we used is the debt and activity transactions of 10,069 Centrelink customers from July 2007 to February 2008. In Centrelink, every single contact (e.g., because of a circumstance change) of a customer may trigger a sequence of activities running. As a result, large volumes of activity based transactions are recorded in an activity transactional database. In the original activity transactional table, each activity has 35 attributes, and we selected four of them which are related to this study. These attributes are "Person ID", "Activity Code", "Activity Date" and "Activity Time", as shown in Table 3. We sorted the activity data according to "Activity Date" and "Activity Time" to construct activity sequences. The debt data consists of "Person ID" and "Debt Transaction Date".

There are 155 different activity codes in the sequences. Different from supermarket basket analysis, every transaction in the application is composed of one activity only. The activities in four months before a debt were believed by domain experts to be related to the debt occurrence. If there were no debts for a customer during the period from July 2007 to February 2008, the activities in the first four months were taken as a sequence associated with no debts. After data cleaning and preprocessing, there are 15,931 sequences constructed with 849,831 activity records in this case study.

Table 4. Selected Positive and Negative Sequential Rules

Type	Rule	Support	Confidence	Lift
I	REA ADV ADV→DEB	0.103	0.53	2.02
	DOC DOC REA REA ANO→DEB	0.101	0.33	1.28
	RPR ANO→DEB	0.111	0.33	1.25
	RPR STM STM RPR→DEB	0.137	0.32	1.22
	MCV→DEB	0.104	0.31	1.19
	ANO→DEB	0.139	0.31	1.19
	STM PYI→DEB	0.106	0.30	1.16
II	STM PYR RPR REA RPT→ ¬DEB	0.166	0.86	1.16
	MND→ ¬DEB	0.116	0.85	1.15
	STM PYR RPR DOC RPT→ ¬DEB	0.120	0.84	1.14
	STM PYR RPR REA PLN→ ¬DEB	0.132	0.84	1.14
	REA PYR RPR RPT→ ¬DEB	0.176	0.84	1.14
	REA DOC REA CPI→ ¬DEB	0.083	0.83	1.12
	REA CRT DLY→ ¬DEB	0.091	0.83	1.12
	REA CPI→ ¬DEB	0.109	0.83	1.12
III	¬{PYR RPR REA STM}→DEB	0.169	0.33	1.26
	¬{PYR CCO}→DEB	0.165	0.32	1.24
	¬{STM RPR REA RPT}→DEB	0.184	0.29	1.13
	¬{RPT RPR REA RPT}→DEB	0.213	0.29	1.12
	¬{CCO RPT}→DEB	0.171	0.29	1.11
	¬{CCO PLN}→DEB	0.187	0.28	1.09
	¬{PLN RPT}→DEB	0.212	0.28	1.08
IV	¬{ADV REA ADV}→ ¬DEB	0.648	0.80	1.08
	¬{STM EAN}→ ¬DEB	0.651	0.79	1.07
	¬{REA EAN}→ ¬DEB	0.650	0.79	1.07
	¬{DOC FRV}→ ¬DEB	0.677	0.78	1.06
	¬{DOC DOC STM EAN}→ ¬DEB	0.673	0.78	1.06
	¬{CCO EAN}→ ¬DEB	0.681	0.78	1.05

3.2 Results of Negative Sequential Pattern Mining

Our previous technique on *negative sequential rules* [28] was used to find both positive and negative sequential patterns from the above data. By setting the minimum support to 0.05, that is, 797 out of 15,931 sequences, 2,173,691 patterns were generated and the longest pattern has 16 activities. From the patterns, 3,233,871 positive and negative rules were derived. Some selected sequential rules are given in Table 4, where "DEB" stands for debt and the other codes are activities. The rules marked by "Type I" are positive sequential rules, while others are negative ones.

3.3 Evaluation of Sequence Classification

The performance of the classifiers using both positive and negative sequential patterns were tested and compared with the classifiers using positive patterns only.

In the discovered rules shown in Table 4, generally speaking, Type I rules are positive patterns and all the other three types are negative ones. However, in the binary classification problem in our case study, $A \rightarrow \neg DEB$ can be taken

as a positive rule $A \rightarrow c_2$, where c_2 denotes "no debt". Therefore, we treated Type I and Type II patterns as positive and Type III and Type IV as negative. That is, in the results shown in Tables 6–9, the traditional classifiers (labelled as "Positive") were built using both Type I and II rules, while our new classifiers (labelled as "Neg& Pos") were built using all four types of rules. However, in applications where there are multiple classes, Type II rules are negative.

By setting the minimum support to 0.05 and 0.1, respectively, we got two sets of sequential patterns, "PS05" and "PS10". The numbers of the four types of patterns are shown in Table 5. There are 775,175 patterns in "PS10" and 3,233,871 patterns in "PS05". It is prohibitively time consuming to do coverage test and build classifiers on so large sets of patterns. In this experiment, we ranked the patterns according to W_s. Then, we extracted the top 4,000 and 8,000 patterns from "PS05" and "PS10" and referred to them as "PS05-4K", "PS05-8K", "PS10-4K" and "PS10-8K", respectively.

After that, two groups of classifiers were built. The first group, labelled as "Neg& Pos", were built with both negative and positive patterns (i.e., all four types of rules), and the other group, labelled as "Positive", were built with positive patterns (i.e., Type I and II rules) only. In order to compare the two groups of classifiers, we selected various numbers of patterns from the ones passing coverage test to build the final classifiers and the results are shown in Tables 6–9. In the four tables, the first rows show the number of patterns used in the classifiers. In Tables 8 and 9, some results are not available for pattern number as 200 and 300, because there are less than 200 (or 300) patterns remaining after coverage test.

From the four tables, we can see that, if built with the same number of rules, in terms of recall, our classifiers built with both positive and negatives rules outperforms traditional classifiers with only positive rules under most conditions. It means that, with negative rules involved, our classifiers can predict more debt occurrences.

As shown by the results on "PS05-4K" in Table 6, our classifiers is superior to traditional classifiers with 80, 100 and 150 rules in recall, accuracy and precision.

From the results on "PS05-8K" shown in Table 7, we can see that our classifiers with both positive and negatives rules outperforms traditional classifiers with only positive rules in accuracy, recall and precision in most of our experiments. Again, it also shows that the recall is much improved when negative rules are involved.

As shown by Tables 8 and 9, our classifiers have higher recall with 80, 100 and 150 rules. Moreover, our best classifier is the one with 60 rules, which has accuracy=0.760, specificity=0.907 and precision=0.514. It is better in all the three measures than all traditional classifiers given in the two tables.

One interesting thing we found is that, the number of negative patterns used for building our classifiers is very small, compared with that of positive patterns (see Table 10). Especially for "PS05-4K" and "PS05-8K", the two pattern sets chosen from the mined patterns with minimum support=0.05, there are respectively only 4 and 7 negative patterns used in the classifiers. However, these

Table 5. The Number of Patterns in PS10 and PS05

	PS10 ($min\_sup = 0.1$)		PS05 ($min\_sup = 0.05$)	
	Number	Percent(%)	Number	Percent(%)
Type I	93,382	12.05	127,174	3.93
Type II	45,821	5.91	942,498	29.14
Type III	79,481	10.25	1,317,588	40.74
Type IV	556,491	71.79	846,611	26.18
Total	775,175	100	3,233,871	100

Table 6. Classification Results with Pattern Set PS05-4K

Pattern Number		40	60	80	100	150	200	300
Neg&Pos	Recall	.438	.416	**.286**	**.281**	**.422**	.492	.659
	Precision	.340	.352	.505	**.520**	**.503**	.474	.433
	Accuracy	.655	.670	**.757**	**.761**	**.757**	.742	.705
	Specificity	.726	.752	.909	.916	.865	.823	.720
Positive	Recall	.130	.124	.141	.135	.151	.400	.605
	Precision	.533	.523	.546	.472	.491	.490	.483
	Accuracy	.760	.758	.749	.752	.754	.752	.745
	Specificity	.963	.963	.946	.951	.949	.865	.790

several negative patterns do make a difference when building classifiers. Three examples of them are given as follows.

- $\neg ADV \rightarrow \neg DEB$ (CCR=1.99, conf=0.85)
- $\neg(STM, REA, DOC) \rightarrow \neg DEB$ (CCR=1.86, conf=0.84)
- $\neg(RPR, DOC) \rightarrow \neg DEB$ (CCR=1.71, conf=0.83)

Some examples of other rules used in our classifiers are

- $STM, RPR, REA, EAD \rightarrow DEB$ (CCR=18.1)
- $REA, CCO, EAD \rightarrow DEB$ (CCR=17.8)
- $CCO, MND \rightarrow \neg DEB$ (CCR=2.38)

4 Related Work

Our study is related to the previous work on negative sequential pattern mining, sequence classification and fraud/intrusion detection. In this section, we review the related work briefly.

Table 7. Classification Results with Pattern Set PS05-8K

Pattern Number		40	60	80	100	150	200	300
Neg&Pos	Recall	**.168**	**.162**	**.205**	**.162**	**.173**	**.341**	**.557**
	Precision	**.620**	**.652**	**.603**	.625	**.615**	**.568**	.512
	Accuracy	**.771**	**.774**	**.773**	**.771**	**.771**	**.775**	**.762**
	Specificity	.967	.972	.956	.969	.965	.916	.829
Positive	Recall	.141	.103	.092	.092	.108	.130	.314
	Precision	.542	.576	.548	.548	.488	.480	.513
	Accuracy	.761	.762	.760	.760	.754	.753	.760
	Specificity	.962	.976	.976	.976	.963	.955	.904

Table 8. Classification Results with Pattern Set PS10-4K

Pattern Number		40	60	80	100	150
Neg&Pos	Recall	0	.303	**.465**	**.535**	**.584**
	Precision	0	**.514**	.360	.352	.362
	Accuracy	.756	**.760**	.667	.646	.647
	Specificity	1	**.907**	.733	.682	.668
Positive	Recall	.373	.319	.254	.216	.319
	Precision	.451	.421	.435	.430	.492
	Accuracy	.736	.727	.737	.738	.753
	Specificity	.853	.858	.893	.907	.893

Table 9. Classification Results with Pattern Set PS10-8K

Pattern Number		40	60	80	100	150	200
Neg&Pos	Recall	0	.303	**.465**	**.535**	**.584**	N/A
	Precision	0	**.514**	.360	.352	.362	N/A
	Accuracy	.756	**.760**	.667	.646	.647	N/A
	Specificity	1	**.907**	.733	.682	.668	N/A
Positive	Recall	.459	.427	.400	.378	.281	.373
	Precision	.385	.397	.430	.438	.464	.500
	Accuracy	.688	.701	.724	.729	.745	.756
	Specificity	.762	.790	.829	.843	.895	.879

Table 10. The Number of Patterns in the Four Pattern Sets

Pattern Set	PS10-4K	PS10-8K	PS05-4K	PS05-8K
Type I	2,621	5,430	1,539	1,573
Type II	648	1,096	2,457	6,420
Type III	2	5	0	0
Type IV	729	1,469	4	7
Total	4,000	8,000	4,000	8,000

4.1 Negative Sequential Pattern Mining

Since sequential pattern mining was first proposed in [1], a few sequential methods have been developed, such as GSP (Generalized Sequential Patterns) [19], FreeSpan [8], PrefixSpan [17], SPADE [26] and SPAM [2]. Most of the sequential pattern mining algorithms focus on the patterns appearing in the sequences, i.e., the positively correlated patterns. However, the absence of some items in sequences may also be interesting in some scenarios. For example, in social welfare, the lack of follow-up examination after the address change of a customer may result in overpayment to him/her. Such kind of sequences with the non-occurrence of elements are negative sequential patterns. Only several studies look at this issue.

Sun *et al.* [20] proposed negative event-oriented patterns in the form of $\neg P \xrightarrow{T} e$, where e is a target event, P is a negative event-oriented pattern, and the occurrence of P is unexpectedly rare in T-sized intervals before target events. P is supposed to be an "existence pattern" (i.e., a frequent itemset without time order), instead of a sequential pattern, though it is claimed that the discussion can be extended to sequential patterns.

Bannai *et al.* [3] proposed a method for finding the optimal pairs of string patterns to discriminate between two sets of strings. The pairs are in the form of $p' \wedge q'$ or $p' \vee q'$, where p' is either p or $\neg p$, q' is either q or $\neg q$, and p and q are two substrings. Their concern is whether p and q appear in a string s.

Ouyang and Huang [16] proposed the notion of negative sequences as $(A, \neg B)$, $(\neg A, B)$ and $(\neg A, \neg B)$. Negative sequential patterns are derived from infrequent sequences. A drawback is that both frequent and infrequent sequences have to be found at the first stage, which demands a large amount of space.

Lin *et al.* [14] designed an algorithm NSPM (Negative Sequential Patterns Mining) for mining negative sequential patterns. In their negative patterns, only the last element can be negative, and all other elements are positive.

4.2 Sequence Classification

Classification on sequence data is an important problem. A few methods have been developed.

Wu *et al.* [23] proposed a neural network classification method for molecular sequence classification. The molecular sequences are encoded into input vectors of a neural network classifier, by either an n-gram hashing method or a SVD (Singular Value Decomposition) method.

Chuzhanova *et al.* [6] proposed to use Gamma (or near-neighbour) test to select features from l-grams over the alphabet. The method was used to classify the large subunits rRNA, and the nearest-neighbour criterion was used to estimate the classification accuracy based on the selected features.

Lesh *et al.* [11] used sequential patterns as features in classification. Sequence mining is first employed to find sequential patterns correlated with the target classes, and then the discovered patterns are used as features to build classifiers with standard classification algorithms, such as Naïve Bayes. Their experimental

results show that using sequential patterns as features can improve the accuracy substantially. Compared to our work, they did not consider negative sequential patterns.

Tseng and Lee [21] designed algorithm CBS (Classify-By-Sequence) for classifying large sequence data sets. Sequential pattern mining and probabilistic induction are integrated for efficient extraction of sequential patterns and accurate classification.

Li and Sleep [12] proposed a robust approach to sequence classification, where n-grams of various lengths are used to measure the similarity between sequences, a modified LZ78 algorithm is employed for feature selection, and a Support Vector Machine (SVM) is used as the classifier.

A discriminatively trained Markov Model (MM(k-1)) for sequence classification was proposed by Yakhnenko *et al.* [25]. Their experimental results show that their classifiers are comparable in accuracy and more efficient than Support Vector Machines trained by k-gram representations of sequences.

Lei and Govindaraju [10] proposed to use an intuitive similarity measure, ER^2, for multi-dimensional sequence classification based on SVM. The measure is used to reduce classification computation and speed up the decision-making of multi-class SVM.

Exarchos *et al.* [7] proposed a two-stage methodology for sequence classification based on sequential pattern mining and optimization. In the first stage, sequential pattern mining is used and a sequence classification model is built based on the extracted sequential patterns. Then, weights are applied to both sequential patterns and classes. In the second stage, the weights are tuned with an optimization technique to achieve optimal classification accuracy.

Xing *et al.* [24] studied the problem of early prediction using sequence classifiers. The prefix of a sequence as short as possible is used to make a reasonably accurate prediction. They proposed a sequential classification rule method to mine sequential classification rules, which are then selected by an early-prediction utility measure. Based on the selected rules, a generalized sequential decision tree method is used to build a classification model with a divide-and-conquer strategy.

In all the above studies, no negative sequential patterns are considered.

4.3 Fraud/Intrusion Detection

Some applications similar to debt detection are fraud detection, terrorism detection, financial crime detection, network intrusion detection and spam detection. Different from transactional fraud detection which attempts to classify a transaction or event as being legal or fraud, our techniques try to predict the likelihood of a customer being fraud based on his past activities. It is at customer level instead of transaction level.

Bonchi *et al.* [4] proposed a classification-based methodology for planning audit strategies in fraud detection and presented a case study on illustrating how classification techniques can be used to support the task of planning audit strategies. The models are constructed by analysing historical audit data. Then, the

models are used to plan effectively future audits for the detection of tax evasion. A decision tree algorithm, $C5.0$, was used in their case study. Although the target problem is similar to ours, the data used is different. We used transactional data which records activities related to customers. Because the time order in activities is important for predicting debt occurrences, sequence classifiers instead of decision trees are used in our application.

Rosset *et al.* [18] studied the fraud detection in telecommunication and presented a two-stage system based on $C4.5$ to find fraud rules. They adapted the $C4.5$ algorithm for generating rules from bi-level data, i.e., customer data and behaviour-level data. However, the behaviour data they used is the statistics in a short time frame, such as the number of international calls and total duration of all calls in a day, which is different from the sequential patterns in our techniques.

Julisch and Dacier [9] used techniques of episode rules and conceptual clustering to mine historical alarms for network intrusion detection. Their episode rules are designed to predict the occurrence of certain alarms based on other alarms. Negative sequential patterns are not taken into account in their model.

5 Conclusions and Discussion

We presented a new technique for building sequence classifiers with both positive and negative sequential patterns. We also presented an application for debt detection in the domain of social security, which shows the effectiveness of the proposed technique.

A limitation of our proposed technique is that an element in a sequence is assumed to be a single event, which is based on the transaction data in this application in social security. However, in other applications, an element may be composed of multiple items. Therefore, to extend our techniques to such general sequence data will be part of our future work.

Another limitation is that time constraints are only partly considered in our techniques. What we did is setting the time window so that a pattern is less than 4 months, based on domain experts' suggestions. Nevertheless, we have not set any other time constraints, such as the time interval between adjacent elements. In other applications, it may be interesting to find patterns with the above constraints and use them to build sequence classifiers.

A third limitation is that, in real world applications, there are different costs with correct predictions, false positives and false negatives, and it will be more fair and more useful when measuring the performance of classifiers by taking the above costs into consideration. We are currently in the progress of the above work.

In our future work, we will also use time to measure the performance of our classifiers, because it is desirable in real-world applications to predict debt occurrences as early as possible. Using time to measure the utility of negative patterns and to build sequence classifiers for early detection will be part of our future work.

Last but not least, patterns in data may keep changing as time goes on, and the learned patterns and the built classifiers may become out-of-date. New labelled data (e.g., new debts) from oncoming data can be used to improve the classifiers. Therefore, it is imperative to build an adaptive online classifier which can adapt itself to the changes in new data.

Acknowledgments

This work was supported by the Australian Research Council (ARC) Linkage Project LP0775041 and Discovery Projects DP0667060 & DP0773412, and by the Early Career Researcher Grant from University of Technology, Sydney, Australia.

We would like to thank Mr. Peter Newbigin and Mr. Brett Clark from Business Integrity Review Operations Branch, Centrelink, Australia for their support of domain knowledge and helpful suggestions.

References

1. Agrawal, R., Srikant, R.: Mining sequential patterns. In: Proc. of the 11th International Conference on Data Engineering, Taipei, Taiwan, 1995, pp. 3–14. IEEE Computer Society Press, Los Alamitos (1995)
2. Ayres, J., Flannick, J., Gehrke, J., Yiu, T.: Sequential pattern mining using a bitmap representation. In: KDD 2002: Proc. of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 429–435. ACM, New York (2002)
3. Bannai, H., Hyyro, H., Shinohara, A., Takeda, M., Nakai, K., Miyano, S.: Finding optimal pairs of patterns. In: Jonassen, I., Kim, J. (eds.) WABI 2004. LNCS (LNBI), vol. 3240, pp. 450–462. Springer, Heidelberg (2004)
4. Bonchi, F., Giannotti, F., Mainetto, G., Pedreschi, D.: A classification-based methodology for planning audit strategies in fraud detection. In: Proc. of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, CA, USA, pp. 175–184. ACM Press, New York (1999)
5. Centrelink. Centrelink annual report 2004-2005. Technical report, Centrelink, Australia (2005)
6. Chuzhanova, N.A., Jones, A.J., Margetts, S.: Feature selection for genetic sequence classification. Bioinformatics 14(2), 139–143 (1998)
7. Exarchos, T.P., Tsipouras, M.G., Papaloukas, C., Fotiadis, D.I.: A two-stage methodology for sequence classification based on sequential pattern mining and optimization. Data and Knowledge Engineering 66(3), 467–487 (2008)
8. Han, J., Pei, J., Mortazavi-Asl, B., Chen, Q., Dayal, U., Hsu, M.-C.: Freespan: frequent pattern-projected sequential pattern mining. In: KDD 2000: Proc. of the 6th ACM SIGKDD international conference on Knowledge discovery and data mining, Boston, Massachusetts, USA, pp. 355–359. ACM, New York (2000)
9. Julisch, K., Dacier, M.: Mining intrusion detection alarms for actionable knowledge. In: Proc. of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Edmonton, Alberta, Canada, pp. 366–375. ACM, New York (2002)

10. Lei, H., Govindaraju, V.: Similarity-driven sequence classification based on support vector machines. In: ICDAR 2005: Proc. of the 8th International Conference on Document Analysis and Recognition, Washington, DC, USA, 2005, pp. 252–261. IEEE Computer Society, Los Alamitos (2005)
11. Lesh, N., Zaki, M.J., Ogihara, M.: Mining features for sequence classification. In: KDD 1999: Proc. of the 5th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 342–346. ACM, New York (1999)
12. Li, M., Sleep, R.: A robust approach to sequence classification. In: ICTAI 2005: Proc. of the 17th IEEE International Conference on Tools with Artificial Intelligence, Washington, DC, USA, pp. 197–201. IEEE Computer Society, Los Alamitos (2005)
13. Li, W., Han, J., Pei, J.: Cmar: Accurate and efficient classification based on multiple class-association rules. In: ICDM 2001: Proc. of the 2001 IEEE International Conference on Data Mining, Washington, DC, USA, pp. 369–376. IEEE Computer Society, Los Alamitos (2001)
14. Lin, N.P., Chen, H.-J., Hao, W.-H.: Mining negative sequential patterns. In: Proc. of the 6th WSEAS International Conference on Applied Computer Science, Hangzhou, China, pp. 654–658 (2007)
15. Liu, B., Hsu, W., Ma, Y.: Integrating classification and association rule mining. In: KDD 1998: Proc. of the 4th International Conference on Knowledge Discovery and Data Mining, pp. 80–86. AAAI Press, Menlo Park (1998)
16. Ouyang, W., Huang, Q.: Mining negative sequential patterns in transaction databases. In: Proc. of 2007 International Conference on Machine Learning and Cybernetics, Hong Kong, pp. 830–834. China (2007)
17. Pei, J., Han, J., Mortazavi-Asl, B., Pinto, H., Chen, Q., Dayal, U., Hsu, M.-C.: Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth. In: ICDE 2001: Proc. of the 17th International Conference on Data Engineering, Washington, DC, USA, pp. 215–224. IEEE Computer Society, Los Alamitos (2001)
18. 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, San Diego, CA, USA, August 1999, pp. 409–413 (1999)
19. Srikant, R., Agrawal, R.: Mining sequential patterns: Generalizations and performance improvements. In: Apers, P.M.G., Bouzeghoub, M., Gardarin, G. (eds.) EDBT 1996. LNCS, vol. 1057, pp. 3–17. Springer, Heidelberg (1996)
20. Sun, X., Orlowska, M.E., Li, X.: Finding negative event-oriented patterns in long temporal sequences. In: Dai, H., Srikant, R., Zhang, C. (eds.) PAKDD 2004. LNCS (LNAI), vol. 3056, pp. 212–221. Springer, Heidelberg (2004)
21. Tseng, V.S.-M., Lee, C.-H.: Cbs: A new classification method by using sequential patterns. In: SDM 2005: Proc. of the 2005 SIAM International Data Mining Conference, Newport Beach, California, USA, pp. 596–600 (2005)
22. Verhein, F., Chawla, S.: Using significant, positively associated and relatively class correlated rules for associative classification of imbalanced datasets. In: ICDM 2007: Proc. of the 7th IEEE International Conference on Data Mining, pp. 679–684 (2007)
23. Wu, C.H., Berry, M.W., Fung, Y.-S., McLarty, J.: Neural networks for molecular sequence classification. In: Proc. of the 1st International Conference on Intelligent Systems for Molecular Biology, pp. 429–437. AAAI Press, Menlo Park (1993)
24. Xing, Z., Pei, J., Dong, G., Yu, P.: Mining sequence classifiers for early prediction. In: SDM 2008: Proc. of the 2008 SIAM international conference on data mining, Atlanta, GA, USA, April 2008, pp. 644–655 (2008)

25. Yakhnenko, O., Silvescu, A., Honavar, V.: Discriminatively trained markov model for sequence classification. In: ICDM 2005: Proc. of the 5th IEEE International Conference on Data Mining, Washington, DC, USA, pp. 498–505. IEEE Computer Society, Los Alamitos (2005)
26. Zaki, M.J.: Spade: An efficient algorithm for mining frequent sequences. Machine Learning 42(1-2), 31–60 (2001)
27. Zhao, Y., Zhang, H., Cao, L., Zhang, C., Bohlscheid, H.: Efficient mining of event-oriented negative sequential rules. In: WI 2008: Proc. of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence, Sydney, Australia, December 2008, pp. 336–342 (2008)
28. Zhao, Y., Zhang, H., Cao, L., Zhang, C., Bohlscheid, H.: Mining both positive and negative impact-oriented sequential rules from transactional data. In: Proc. of the 13th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2009), Bangkok, Thailand, April 2009, pp. 656–663 (2009)