Faculty of Engineering and Information Technology

University of Technology, Sydney

SOCIAL SECURITY DATA MINING: AN AUSTRALIAN CASE STUDY

A Thesis Submitted in Fulfilment of the Requirements for The Degree of Doctor of Philosophy, Faculty of Engineering and Information Technology, University of Technology Sydney.

By

Hans Michael Bohlscheid

October 2013

CERTIFICATE OF AUTHORSHIP/ORIGINALITY

I, Hans Michael Bohlscheid, certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of Requirements for a degree except as fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Signature of Candidate

____

Hans M Bohlscheid

01 /10 /2013

ACKNOWLEDGEMENT

First and foremost, I would like to express my deepest gratitude to Professor Longbing Cao for his supervision and ongoing support over many years. I regard Professor Cao as an energetic, inspiring individual and will always remember our many discussions and his spontaneous ideas.

I also remain grateful to my colleagues and friends Drs Huaifeng Zhang and Yanchang Zhao for their advice regarding the technical aspects of my thesis.

In the mix of the many people to whom I am indebted are Yuri Zubrutsky, Daniel Marlay and Richard Brookes who independently evaluated my results and last but not least, I extend my heartfelt thanks to my dear colleagues Peter Newbigin and Brett Clark for their expertise in data extraction and their acute knowledge of Centrelink business practices.

In their own unique way, each of the above contributed to the enjoyment and success of my research and collectively, they are responsible for the skills and subject matter knowledge I possess today.

> *"I was striking an uneasy balance between the ambition I had for myself, and what those closest to me expected of me. So I stopped pretending to myself that I was anything other than what I was, and began to direct all my energy into finishing the only work that mattered to me."*
>
> *J.K. Rowling*

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Abstract

Data mining in business applications has become an increasingly recognized and accepted area of enterprise data mining in recent years. In general, while the general principle and methodologies of data mining and machine learning are applicable for any business applications, it is often essential to develop specific theories, tools and systems for mining data in a particular domain such as social security and social welfare business. This necessity has led to the concept of *social security and social welfare data mining*, the focus of this thesis work.

Social security and social welfare business involves almost every citizen's life at different life periods. It provides fundamental and crucial government services and support to varied populations of specific need. A typical scenario in Australia is that it not only connects one third of our populations, but also associates with many relevant stakeholders, including banking business, taxation and Medicare. Such business engages complicated infrastructure, networks, mechanisms, policies, activities, and transactions. Data mining of such business is a brand new application area in the data mining community.

Mining such social welfare business and data is challenging. The challenges come from the unavailable benchmark and experience in the data mining for this particular domain, the complexities of social welfare business and data, the exploration of possible doable tasks, and the implementation of data mining techniques in relation to the business objectives.

In this thesis, which adopts a practice-based innovative attitude and focusses on the marriage of social welfare business with data mining, we believe we have realised our objective of providing a systematic and comprehensive overview of the social security and social welfare data mining. The main contributions consist of the following aspects:

- As the first work of its kind, to the best of our knowledge, we present an overall picture of social security and social welfare data mining, as a new domain driven data mining application.
- We explore the business nature of social security and social welfare, and the characteristics of social security data.

- We propose a concept map of social security data mining, catering for main complexities of social welfare business and data, as well as providing opportunities for exploring new research issues in the community.
- Several case studies are discussed, which demonstrate the technical development of social security data mining, and the innovative applications of existing data mining techniques.

The nature of social welfare is spreading widely across the world in both developed and developing countries. This thesis work therefore is timely and could be of important business and government value for better understanding our people, our policies, our objectives, and for better services of those people of genuine needs.

1 Introduction

1.1 Background

Machine learning and data mining are increasingly used in business applications [21], and in particular, in public sectors [94]. A distinct public-sector area is social security and social welfare [121] which suffers critical business problems, such as the loss of billions of dollars in annual service delivery because of fraud and incorrect payments [121], [134]. People working in different communities are increasingly interested in "what do social security data show" [93] and recognize the value of data-driven analysis and evidence-based decision-making to enhance public service objectives, payment accuracy, and compliance [101], [121]. Within the marriage of machine learning and data mining with public sectors, an emerging data mining area is the *analysis of social security/welfare business and data* [16].

Mining social security/welfare data is challenging [16] in that the data are very complex, involving all the major issues that are discussed in the data quality and engineering field, such as sparseness, behaviors, dynamics, and distribution. Key aspects contributing to challenges in mining social security data are many [16], e.g., 1) specific business objectives in social security and government objectives, 2) specific business processes and outcomes, 3) heterogeneous data sources, 4) interactions between customers and government officers, 5) customer behavioral dynamics, and 6) general challenges in handling enterprise data, such as data imbalance, rare events, high dimension, and so on.

Studies on social security issues [16] started in the middle of the 20th century [30], [35]. Since then, many researchers have worked on different topics. The majority of research has been conducted from political [44], [55], [58], [60], [73], economic [7], [11], [30], [35], [67], [91], [95], [98], [109], sociological [58], [59], and regional [2], [10], [29], [43], [52], [60], [70], [75], [85], [87], [104] perspectives, compared with a much delayed effort made on the technical aspects [23], [64], [68], [83], [94]. The main issues involve problem analysis, process and policy modeling, business analysis, correlation analysis, infrastructure development, and emerging data-driven analysis. In contrast with the dominant fact and trend of policy and economy oriented studies, very limited research [23], [39], [68], [69], [103],

[105], [106], [107], [108], [110], [111], [112], [113], [114], [115], [116] can be found in the literature on mining social security and social welfare data.

Social security data mining (SSDM) [16] seeks to discover interesting patterns and exceptions in social security and social welfare data. From the data mining goal perspective, it aims to handle different business objectives, such as debt prevention. From the data mining task perspective, it involves both traditional data mining methods, such as classification, as well as the need to invent advanced techniques, e.g., complex sequence analysis. Australia is one of the most developed social welfare countries in the world in terms of government policies, infrastructure, the population of benefit recipients, and the advancement of social security techniques and tools [121]. Since 2004, SSDM has been initiated with Centrelink for the Australian Commonwealth Government through a series of projects. We have developed models, algorithms, and systems to identify key drivers, factors, patterns, and exceptions, indicating high risk of customers, customer circumstance changes, declarations, and interactions between customers and government officers. The findings have proven to be very useful for overpayment prevention, recovery, prediction, and deep understanding of customer activities and intervention, which involve the recovery and prevention of overpayments (also called "debt" when referring to the part of payments to which the recipient is not entitled) for the government. These substantial practices have been selected as one of the IEEE's International Conference on Data Mining Top 10 Data Mining Case studies [16].

1.2 Business Needs and Research Issues

1.2.1 Business Needs：Social Security Service Delivery, Reform and Challenge

The delivery of Public Sector welfare services in Australia has always represented an enormous task. Traditionally Australian Public Servants saw it as their role to provide citizens with services that were fair, equitable, correctly and lawfully delivered where entitlements under the law were economically and carefully administered with an emphasis on due process.

By the turn of the century, strong Government imperatives and more advanced technology had facilitated the creation of the one-stop shops, and the delivery of electronic services across the Internet had become a reality.

From the 1990s successive governments have sought to make Public Services more responsive to the needs of the citizen or customer, and to increase efficiency and effectiveness in service delivery.

Over the next decade, surveys reported in papers such as the Australian Public Service (APS) Commission's 2001-02 State of the Service, conclude that the APS has been largely successful in establishing a customer focus and improving the standard of service delivery.

In the early 2000s, key initiatives for improving Government service delivery and meeting customer and community expectations included:

- improved access to services;
- commitment to internal and external customer service charters;
- closer working partnerships between policy departments and service delivery agencies;
- greater involvement with other non-Government sector organisations; and
- market research into customer satisfaction and service improvement.

Governments are searching for new ways to find solutions to problems that cross all levels of governments, jurisdictions and portfolios. The challenge is to develop mechanisms, structures and cultures, which facilitate whole of Government approaches that become a characteristic of the way Government works in Australia. And to meet this challenge, Public Sector leaders will need to implement one of their key Public Service Act responsibilities, which is to promote cooperation with other Agencies.

In response to this challenge, in early 2009 Government, industry and community agencies commenced working together to develop innovative changes to payment and service delivery processes. As a result, the Commonwealth Government called for a major reform of the Department of Human Services. In addition to other elements, this entailed the introduction of sophisticated analytics to redesign service delivery in order to maximise convenience for all Australians. That is, giving people more control, better support, and individual assistance when they need it most. Today, after making the following announcement in late 2009, major reforms to service delivery and welfare agencies are well under way:

> *"The Government needs to adapt to current and future challenges facing the nation, including changes in technology, the growing demand on our service delivery agencies and demographic challenges such as Australia's ageing population. The restructure (to the Department of Human Services) comes as recent (data mining) pilot figures show that the Government could have saved $344.1 million over the past three years if the results of data analytics (in Centrelink) had of been applied to look for (process) errors and crack down on fraud"*
>
> *The Minister for Human Services, Chris Bowen MP 17/12/09*

1.2.2 Research Issues in Social Security Data Mining

The data mining technology will definitely be helpful to improve social security service by customer activity analysis, risk analysis, change detection, fraud detection, debt analysis, payment accuracy analysis and so on. Those analysis tasks are comprehensive, involving many aspects of both traditional and emerging data mining techniques. The key challenges are listed as following in terms of social security data processing, pattern analysis, and knowledge delivery, which will be described on detail in Chapter 3.

1.2.2.1 Data Processing

The processing of data characteristics in social security business shares many data processing issues within the data mining and machine-learning community. In particular, the following areas are especially important in SSDM.

1) The processing of activities and activity sequences need to address complex features, such as temporal, spatial, structural and semantic dimensions.
2) Changes are widely dispersed in social security data.
3) Customer–government interaction generates intensive activities in the social security business.
4) To engage multiple sources of social security data.
5) Involve large-scale mixed data.
6) Process extremely imbalanced data in a large scale set, designing proper data structures, filtering, and sampling algorithms to prepare both class and item-set imbalanced data.

1.2.2.2 Pattern Analysis

While many traditional and emerging pattern mining methods and pattern types can be applied directly to SSDM, we also observe specific needs emerging from mining social security data.

1) Challenging issues in detecting changes cover many possibilities.
2) Activities are widely seen in social security data, which create a challenge for traditional data mining.
3) *Impact* modeling measures the impact of certain data on business. The nature of the impact, how to measure it and how it is associated with patterns and debt occurrence, is open to investigation.
4) *Impact*-oriented pattern mining identifies patterns that are associated with specific impact, but it is challenging.
5) Complex business issues and the data characteristics require combined mining, which is much more complex than traditional data mining procedure. Combined mining may include combined data mining models, combined data sources, combined patterns, etc.

1.2.2.3 Knowledge Delivery

Delivering knowledge of business interest which can be taken over by business people is not a trivial problem.

1) How to define and measure business performance from subjective and objective perspectives is worthy of research, as well as which business metrics need to be defined for use to generally measure business impacts associated with patterns.
2) Knowledge actionability is the actionability of identified patterns. In the social security area, it is hard to measure knowledge actionability.
3) Enhancing the actionability of identified patterns is not a straightforward task. It is worthwhile to analyze why the discovered knowledge is not actionable, and what actions can be taken to enhance the actionability.
4) Pattern post processing is an important way to enhance knowledge actionability and is applicable to SSDM.

5) Deliverable representation is to convert SSDM findings into business-oriented deliverables and represents deliverables in a business friendly manner. This is actually a challenging issue, which has not been well studied.

1.3 Research Methodology

1.3.1 Positive / Negative Sequential Rules Mining for Debt-oriented Analysis

Whereas traditional association rule mining and sequential pattern mining deal only with the positive relationship between itemsets or events, it might be interesting to examine frequent sequential rules that display a 'non-occurrence' of some items – which is called negative sequential rules.

In Centrelink, for various reasons, customers on benefit payments or allowances sometimes get overpaid and these overpayments lead to debts owed to Centrelink. Debt-oriented negative sequential rules can be used to find the relationship between transactional activity sequences and debt occurrences, and also find the impact of additional activities on debt occurrence. Chapter 4 will illustrate the novel method on SSDM.

1.3.2 Sequence Classification to Predict Debt-related Activities

From a data mining perspective, sequence classification involves building classifiers using sequential patterns which can be used to predict and further prevent debt occurrence based on customer transactional activity sequences.

In finding that all of the existing sequence classification algorithms could not be used into SSDM, the following improvements were considered.

1) Consider the sequential patterns negatively correlated to debt occurrence, which are very important in debt detection.

2) In order to catch up with the pattern variation over time, an adaptive sequence classification method, which is able to include the latest pattern into the classifier, is to be considered to improve the classification performance on dataset of near future.

Chapter 5 will illustrate the above methods in detail.

1.3.3 Combined Mining for Social Security Data

Combined mining is essential in SSDM, which mines for combined patterns by engaging multisource data and complicated social security data.

Mining combined association rules, which are composed of multiple heterogeneous itemsets from different datasets, need to be considered in SSDM. While combined rule pairs and combined rule clusters are to be built from the combined association rules, the combined patterns will provide more interesting knowledge and more actionable results than traditional association rules.

1.4 Thesis Contribution

The contributions of the thesis are:

- A comprehensive literature review of social security data mining and discussion of different categorizations of the related work are described.
- A general framework of the social security data mining is proposed. It discusses the main data mining goals, tasks, and principal challenges in mining patterns in social security data;
- It proposes several new and effective algorithms and tools to handle social security data analysis, including:
    1) Efficient algorithms for discovering event-oriented positive/negative sequential rules, and novel metrics to measure the impact on outcome. See Chapter 4.
    2) A new technique of sequence classification using both positive and negative sequential patterns, see Section 5.1.
    3) A novel method to boost discriminative frequent patterns for sequence classification, which improves the accuracy of classifier. An adaptive sequence classification model upgrades the sequence classification performance on time-varying sequences. See Section 5.2.
    4) A novel hierarchical algorithm for sequence classification, which can reduce the running time while keeping the performance of the classification. See Section 5.3.
    5) A novel notion of combined patterns to extract useful and actionable knowledge from a large amount of learned rules. See Chapter 6.

6) An efficient algorithm to mine class association rules on datasets with multiple imbalanced attributes. See Chapter 7.

- All the above proposed algorithms and strategies are all applied to social security data analysis, which shows the efficiency and effectiveness of the proposed methods in SSDM.

1.5 Thesis Organization

This thesis is organized as follows.

Chapter 1 provides an introduction to SSDM. It describes background, business issues, research issues and our contributions on the area.

Chapter 2 talks about the review on social security/welfare data mining research.

Chapter 3 presents formal and general descriptions of a framework of SSDM.

Chapters 4-7 introduce some novel data mining models which were applied in SSDM, including:

a) Positive/negative social security sequential rules analysis in Chapter 4;
b) Predict debt-related social security activity sequences in Chapter 5;
c) Mining combined social security patterns in Chapter 6;
d) Rare class association rule mining with multiple imbalanced attributes in Chapter 7;

Chapter 9 presents concluding comments and discusses potential future work the conclusion and future work.

2 Review on Social Security/Welfare Data Mining Research

2.1 Introduction

Data mining is becoming an increasingly hot research field, but a large gap remains between the research of data mining and its application in real-world business. Data mining in social security involves the application of techniques such as decision trees, association rules, sequential patterns and combined association rules. Statistical methods such as the chi-square test and analysis of variance are also useful. The data involved includes demographic data, transactional data and time series data and we were confronted with problems such as imbalanced data, business interestingness, rule pruning and multi-relational data. Some related work include association rule mining [135], sequential pattern mining [136], decision trees [137], clustering [138], interestingness measures [139], redundancy removing [140], mining imbalanced data [141,142], emerging patterns [147], multi-relational data mining [144],[145],[146], [148] and distributed data mining [143, 149, 150].

2.2 Social Security Services and Data

2.2.1 Social Security Business

In countries like Australia and Canada, a variety of social welfare allowances/services and social programs are provided by the government to assist people to become self-sufficient and to support those in need. Fig. 2.1 illustrates a cause–effect relationship between customers and government in the social welfare business. A customer lodges an allowance application, which is checked by the government. Payments are arranged on the premise of customer entitlement and policies. The customer is required to declare any changes that may affect payment entitlement. Once a customer declaration is lodged, it will be verified by the government. As a result, customer payments are further verified and adjusted if necessary. In some cases, overpayments to the customer may occur for reasons such as incorrect declaration. The government will seek to recover the debt, and the customer will be requested to pay back such overpayments through repayment arrangements made between the government and the customer. More information about social welfare business can be found on the respective government websites and in reports [121].

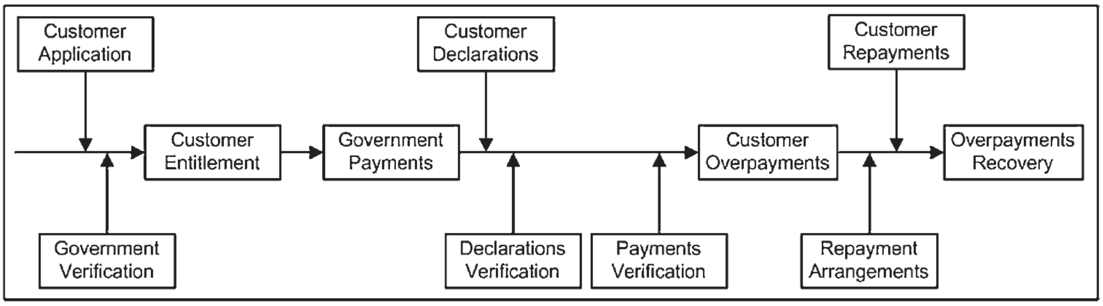


Figure 2-1 Social Security Business Workflow

Taking the Australian social welfare business as an example, as a one-stop-shop, the Australian Commonwealth Department of Human Services (Centrelink) delivers a range of Commonwealth services to the Australian community and is responsible for the distribution of around $86.8 billion, i.e., 30% of the Commonwealth's outlay, to about one-third of Australians [121]. In day-to-day interactions between the government and customers, Centrelink accumulates a large amount of interaction data. For instance, Centrelink provides 361 000 face-to-face services each working day and processes 6.6 billion transactions against customer records each year [122]. It has been shown that the number of such interactions increases every year. The government has progressively recognized the importance of analysing these interactions to obtain a deep understanding of customers and organization–customer relationships, to actively manage customers, to improve government service quality and objectives, and to inform policy design.

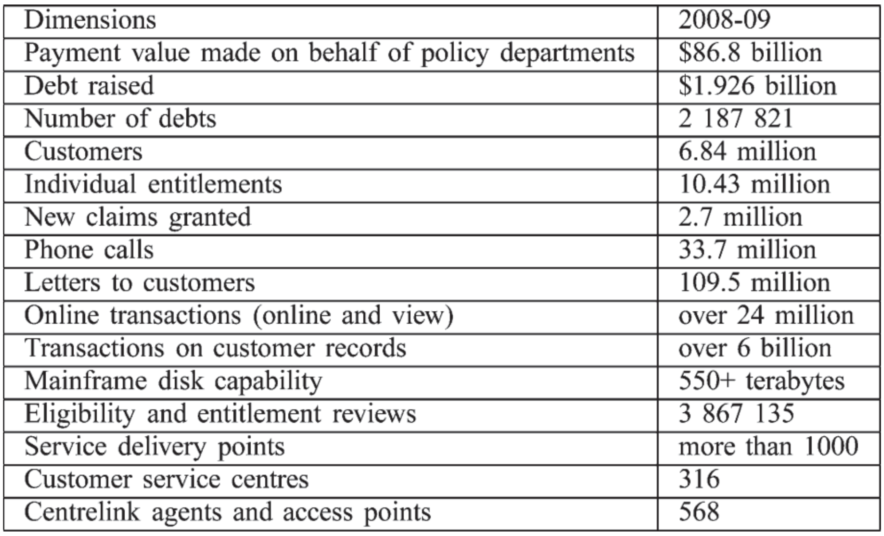
An issue of particular interest is the identification of the drivers that cause noncompliance in organization–customer interactions. Noncompliance drivers may result from many aspects or staff errors. The 2007–2008 audit report by the Australian National Audit Office (ANAO) [119] drew attention to the importance of deeply understanding customers, and of addressing the behaviour and behavioural changes in rising debt from the perspective of the customer, government administration, client group, and community.

2.2.2 Social Security Data

2.2.2.1 Data Categories

As a result of implementing day-to-day social security services, a huge amount of data have been accumulated which increases dramatically every day. Table 2.1 provides an overview of some Centrelink business dimensions related to data [121].

Table 2-1 Centrelink Business Dimensions 2008–2009



Dimensions	2008-09
Payment value made on behalf of policy departments	$86.8 billion
Debt raised	$1.926 billion
Number of debts	2 187 821
Customers	6.84 million
Individual entitlements	10.43 million
New claims granted	2.7 million
Phone calls	33.7 million
Letters to customers	109.5 million
Online transactions (online and view)	over 24 million
Transactions on customer records	over 6 billion
Mainframe disk capability	550+ terabytes
Eligibility and entitlement reviews	3 867 135
Service delivery points	more than 1000
Customer service centres	316
Centrelink agents and access points	568

As Table 2.1 shows, the huge amount of social security data accumulated by Centrelink consist of very useful information recorded from customer service centers, agents and access points, the Internet, interviews and reviews for all services, customers, staff and agents, and debt. The $1926 million in debt raised in 2008–2009 compares with $1831 million in 2007–2008. Such data can be classified into the following categories [16]:

1) Customer demographic and circumstance data, recording information about a customer and his/her circumstances, circumstance changes, etc; for instance, home address and the history of address change;
2) Benefit/allowance data, regarding the information about specific benefit/allowance design and applicability in alignment with customer eligibility, and management processes;

3) Customer pathway data, reflecting the history and relevant details of a customer's use of government services, such as the number of services, when, and from which service centers the services have been applied for, and granted;
4) Activity data, providing activity records information about who (maybe multiple operators) processes what types of activities (say change of address) from where (say customer service centers) and for what reasons (say the action of receipt of source documents) at what time (date and time), as well as the resultant outcomes (say raising or recovering debt) [24], [120];
5) Facility usage data, regarding the resources used by or for customers, e.g., phone calls and online services;
6) Service policy data, information about policies, the applications of policies to customers with particular circumstances;
7) Service transactional data, day-to-day information recorded regarding the use of services, such as new registrations, new claims, debt review, etc.;
8) Service performance data, concerning service quality and performance, such as overpayments and their distribution, how long on average a customer has to queue, general customer satisfaction, etc.;
9) Interaction data about communications between customers and staff, and between staff from different units; for instance, a customer calls Centrelink to report an income update;
10) Operation data about the resources and infrastructure used for day-to-day business, e.g., how many staff hours are spent on payment reviews;
11) Operational performance, concerning the performance of operational expenses and resource use; for instance, the average cost of recovery per dollar of debt, or the effectiveness of reminder letters in terms of solving problems (such as seeking to recover the outlays).

2.2.2.2 Data Characteristics

From the aforementioned summary, we identify social security data as having the following characteristics [16].

1) *Large-scale:* as shown in Table 2.1, huge amounts of data are collected every day and every year.
2) *Mixed structure:* Data incurred by the business consist of all major types, such as numerical, categorical, textual, discrete, continuous, temporal, and sequential.
3) *Distribution:* Data are collected from service centers, the internet, and access points, and recorded in mainframe storage distributed in large centers; customers are distributed everywhere across the country.
4) *Longitude:* Typically, a customer is engaged with a service for quite a while before they are terminated or transferred to another service type.
5) *High dimension:* Data involve multiple dimensions; for instance, there are over 200 types of activity codes reflecting different actions taken in customer–Centrelink interactions.
6) *Multisource:* Different aspects of information data are recorded separately; any one source of data is insufficient to generate a full picture of a particular service.
7) *Sparseness:* The longitudinal data are generated on demand, which is normally random and infrequent; the resulting data are very sparse; for instance, a customer may have accessed a service center two years previously and, later, contacted another office in another place to update a circumstance change, or request a new service.
8) *Imbalance:* Data are not equally distributed, with some being of much higher frequency than others; for instance, outlays only consist of a very small proportion of the overall expenses in Centrelink; customer–government interaction data are not equally distributed, and some activities occur much more frequently, or in more places, than others; Debt-related data only constitute a very small proportion of social security data [23], which forms a class imbalance. In addition, the customer–government interaction activities that are related to debt are composed of a very limited portion of the whole activity set, which gives rise to an item-set imbalance issue.
9) *Divided quality:* It is known that, with data being recorded in divided quality, some data may be missing, or recorded in duplicate.
10) *Variation:* Changes happen everywhere, involving all of the above aspects and data characteristics; in fact, as identified in the ANAO report [119], changes have a critical effect on business integrity and performance stability.

11) *Coupling relationship:* Data entities (objects) and values are often inter-related because of intrinsic business logics in social security. For instance, a debt may be incurred as a result of a wrong declaration of income and address change; a follow-up arrangement activity is made for the customer to pay back the debt once confirmed (called repayment); the customer may either follow the arrangement (refers to the activities arranged by the government) or take other actions to address the repayment. This example shows that objects (i.e., customer, debt, arrangement, and repayment), transactions (regarding customer, debt, arrangements, repayment, etc.), and behaviors from different objects are inter-related.

The aforementioned characteristics are typically aligned with data complexities currently explored in the broad data mining community. They also create additional challenges for existing data mining methods and algorithms when they are deeply engaged in the social security business. In fact, the social security area presents great possibilities to explore typical data mining complexities in one domain. This leads to the need for further research on SSDM tasks and challenges.

2.3 Comprehensive Picture

Research on social security and welfare issues started in the mid-20th century [30]. Since then, broad-based issues have been added to the investigation and can be categorized into the following main streams [16].

1) *Political perspective:* One of the main streams of research investigates the problems, issues, factors, and impact of social security and welfare from public policy [44], [73], social policy [55], [60], administration [89], governance [58], resistance [32], and practice viewpoints [99].
2) *Economic perspective:* Another dominant fact and trend is the exploration of issues and the effect of social security models and factors from the standpoint of econometrics, public economics, and political economy [44]. This involves analysis and discussions about economy [95], earnings [14], [49], [57], rating [80], savings [4], [11], [65], [73], growth [109], privatization [71], reform [55], [56], [98], labor supply [54], [67], multientity relationship analysis [30], [58], [59], [93], and optimal arrangements [7], [35], [53], [91], [92].

3) *Sociological perspective:* Some researchers are concerned about the social effect of social security policies on society, such as lifecycle [74], [93], demographic [1], [11], behavior [84], [90], aging [72], retirement [10], [50], [51], [65], [79], [81], fraud [32], [88], fairness and affordability [76], etc.
4) *Regional perspective:* Researchers from different countries introduce the development of social security in their countries, for instance, Canada [52], India [2], Latin America [70], [85], the U.S. [124], Britain [60], Sweden [29], China [75], [104], Italy [87], Germany [96], France [10], and Europe [43].
5) *Technical perspective:* An emerging trend in social security is the study of technical issues, e.g., infrastructure development [64], knowledge management [83], policy and process modeling, data-driven analysis [23], [68], [69], [103], [105]–[108], [110]–[116], and correlation analysis crossing multiple areas [94], [110].

2.4 Technical Perspective

From the technical perspective, the main issues [16] that have been addressed in the literature focus on several areas, including problem analysis, process and policy modeling, business oriented analysis, correlation analysis, infrastructure support, and data-driven analysis.

1) *Problem analysis:* From time to time, we find papers discussing or debating the issues of reform [56], crisis [6], [13], issues for policies [55], privatization [71], uncertainty [92], optimization [97], fraud [32], [88], and effect on economy [30], society, capital market [31], human resources [90], [93], etc.
2) *Process and policy modeling:* Different approaches, e.g., empirical analysis, time-series analysis, quantitative comparative analysis, and equilibrium analysis [42], have been used and developed to design, simulate, and evaluate policy, pension, benefit [7], process and their effects, as well as their optimization, choice [65], and performance rating [62] including accuracy [45].
3) *Business-oriented analysis:* Key business issues, such as earnings and income, rate, benefit claiming, behavior, retirement, risk, saving, etc., are studied from political, economic, sociological, and technical perspectives.
4) *Correlation analysis:* The relationship between social security and other economic systems have been studied; for instance, the relationship with health affairs, Medicare

5) *Economic Analysis:* examine issues such as [76], [90], taxation [8], [34], stock and market [31], [34], [41], as well as with political structure [30], economic development [30], labor force [54], [67], [96], and human capital [40].
6) *Infrastructure support:* Discussions have been conducted on building IT systems, supporting the analysis of social security data, simulating and optimizing processes, policy, performance rates, etc.
7) *Data-driven analysis:* Recently, the value of data and data driven decisions has been increasingly recognized. Various analysis approaches are being developed to investigate "what do social security data show" [93], e.g., to identify drivers, enablers [83], service patterns [69], linkages [81], demographic [1], behavior [84], change, data measures and composites [123], fraud [88], and risk adjustment [27] from nonrandom selection, stochastic forecast [72], sequential analysis [116], time-series [73], equilibrium analysis [66], data mining, knowledge management [89], micro estimation [50], and e-government [64] aspects.

Modern computer systems have been widely used in the social security and welfare sectors since the earliest period of the computerization age. Currently, the use of computers for e-government service in the developed countries has reached a very advanced and comprehensive level. The research from an e-government perspective in the social security and social welfare sectors can be categorized into four main streams: IT infrastructure, operational support system, business support system, and decision support system.

1) *IT infrastructure:* The infrastructure supporting business processes, networking, data storage, human–computer interaction, etc.
2) *Operational support systems:* Systems supporting operations, such as network inventory, provisioning services, configuring network components, privacy, security management, etc.
3) *Business support systems:* Systems offering business interactions with customers, for allowance and benefit delivery, service profiling, debt management, review management [121], etc.
4) *Decision support systems:* Systems supporting decision making, including business integrity management, business intelligence systems, real-time and historical data

analysis system, risk analysis and management systems, case management system, and decision-making facilities.

2.5 Related Work of Social Security Data Mining

The public sector has also kept “the frontier spirit alive in the computer science community” [94]. In particular, data-driven decision has recently been increasingly recognized as one of the most powerful tools to improve government service objectives. However, mining social security/welfare data is an open, new area in the data mining community. To the best of our knowledge, only two groups have involved SSDM, and a very limited number of relevant publications can be found in the literature. In the following, we discuss the UNC group’s work and address the practices by the UTS group.

In [38], [39], [68], and [69], a case study was conducted on monthly service data and service variations to detect common patterns of welfare services given over time. The study’s authors used a simple sequence analysis method on monthly service administrative databases, which indicates what services were given when, to whom, and for how long. While “common” service procedures can be identified by simply applying a frequent sequence analysis method, it appears that no additional advancement has been made in tackling critical challenges in the data, e.g., mixed transactional data, imbalanced items, and labels. The method only identifies general frequent procedures that are commonsense to business people. No informative and implicit patterns can be identified in this case study. From a business perspective, the identified frequent patterns are not very helpful, since they reflect the actual service arrangements implemented as per policies. Business people want to discover something they do not already know about their business and to develop a deep understanding of why, and how, specific problems face the organization.

In our substantial literature review of SSDM, no additional references have been identified that provide substantial insights for mining social security/welfare data.

2.6 Retrospection on Mining Social Security Data

Our substantial literature review work and practices in Australia [16] reveal the following observations about the existing research on mining social security/welfare data.

1) It is a very open area in terms of applying and conducting pattern/anomaly discovery on social security data (i.e., SSDM). In the very limited work available from the literature, no such systematic work has been done in terms of drawing an overall picture of SSDM from either the business or technical side, nor in addressing the challenges and opportunities in SSDM.
2) According to our experience in conducting SSDM in Australia, social security/welfare business and data consist of comprehensive characteristics and complexities specific to the data mining community which are not comparable with those in many domains. This is reflected through the nature of mixing politics, economy, society, organizational and business processes, and the Internet, as well as the distribution of information sources, business dynamics, customer–government interaction evolution, and integration of business and technical issues. This makes social security/welfare data very challenging and complex to analyze.
3) In fact, social security data/business provides a relatively complete test bed for both existing and emerging research on data mining thanks to the complexities from data to problem modeling and delivery of knowledge. The mixture of several complex aspects makes SSDM even more challenging, as, for instance, in mining impact-oriented behavior patterns in large-scale of data mixing customer demographics, activities, policies, and performance.
4) Specifically, very limited SSDM work has been done in either research or practice, leaving a big gap in relation to increasing business needs. Besides the data/business characteristics common to many other areas, it is worthwhile and highly demanding to explore characteristics and challenges in the marriage of data mining with social welfare business and to systematically explore business problems, research issues, challenges, limitations in directly applying existing data mining outcomes, and opportunities to invent new techniques.
5) While SSDM involves many challenges common to other domains, the mixture of specific business mechanisms with wide data complexities also makes SSDM important and challenging, in aspects such as specifying/customizing and inventing data mining methods and algorithms to effectively analyze social welfare business, for example, processing specific data characteristics and discovering patterns therein.

Since 2004, we have been engaged in data mining for social welfare business. So far, several projects have been conducted which tackle issues, such as analyzing customer earnings [126], profiling debt recovery and verifying changes in earnings declarations [127], investigating relationships between activity sequences and debt occurrences [24], modeling activity impact on debt risk and cost [23], [112], [114], identifying high impact activities/activity sequences on debt occurrences [23], rating customer risk on causing debt [128], fraud detection [129], and so on.

Starting from the understanding of social security business and data, the following chapters present an SSDM framework and address SSDM goals, tasks, and challenges. We also summarize the real-life practices of SSDM in Australia and discuss the extension of SSDM for mining general public-sector data.

3 Social Security Data Mining Framework

3.1 A Basic Framework

Like any other domain, data mining applications in social security are driven by business objectives and underlying data. Based on the introduction of social security business and data in Section 2.2, Fig. 3.1 [16] presents a high-level SSDM framework. It consists of three layers: the data layer, the business objective layer, and the data mining goal layer.

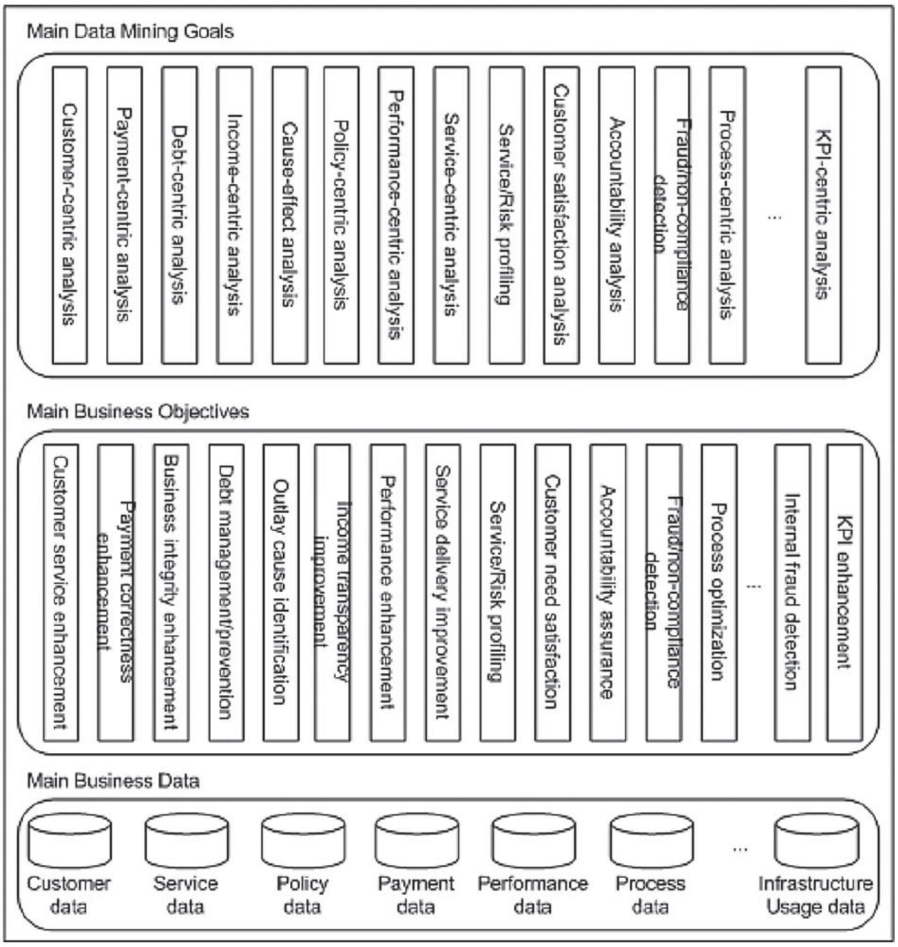


Figure 3-1 SSDM framework

The business objective layer includes the main aims and expectations for the implementation of social security services. For instance, Fig. 3.1 lists the main objectives [121], including customer service enhancement (to instantly provide high-quality services to those with particular needs), payment correctness enhancement (e.g., to pay the right amount to those who are eligible), business integrity enhancement (e.g., to improve the consistency and accuracy and to speed up processing), debt management and prevention (e.g., to recover and prevent debt instantly), outlays cause identification (e.g., to identify outlays incurred by staff error), income transparency improvement (e.g., to improve customer earnings reporting and to detect gray income automatically), performance enhancement (e.g., to reduce customer waiting time in service centers or call centers), service delivery enhancement (e.g., to strip out unnecessary contacts and provide easier and more efficient pathways to services), service/risk profiling (e.g., to identify customers most at risk of incorrect payments and to identify opportunities to reduce the debt more efficiently), customer need satisfaction (e.g., to identify customers with special or more urgent needs than others), accountability assurance (e.g., to identify areas of significant financial or operational risk and to pinpoint more effective arrangements to manage risks), fraud and noncompliance detection (e.g., to identify international or staff fraud and noncompliance), process optimization (e.g., to streamline processes for easier service access and delivery), and key performance indicator (KPI) enhancement (i.e., to identify where and how the key performance indicators can be enhanced).

To support the aforementioned major business objectives, the government invests in efficient information infrastructure. As a result, data are acquired and constantly updated at every place and time in the business operation. The data layer summarizes the main data resources. It consists of customer data (customer demographic and circumstance information), service data (service usage and procedural information), policy data (government policy and the applications of policy), payment data (customer payment information), performance data (service performance and operational performance), process data (business process and change applied to customers), infrastructure usage data (the use of IT resources and services), etc.

While every effort has been made to rectify problems, it has been disclosed that the government is facing longstanding, as well as emerging, problems in achieving and improving the main business objectives [119]. The accumulation of business data provide a unique and essential premise to disclose hidden and implicit channels, indicators, and solutions for these issues, as shown by the data mining pilots in Centrelink. The data mining layer lists the main goals in mining social security data to enhance business objectives; for instance, customer-centric analysis, payment-centric analysis, debt-centric analysis, income-centric analysis, cause-effect analysis, policy-centric analysis, performance-centric analysis, service-centric analysis, service/risk profiling, customer satisfaction analysis, accountability analysis, fraud/noncompliance detection, process-centric analysis, and KPI-centric analysis. These processes will be discussed further in the following sections.

3.2 Social Security Data Mining Goals

We summarize the main data mining goals in the social security area into the following five classes, according to our understanding and practices of key entities, problems, and challenges in social welfare business by data mining: 1) overpayment centric analysis; 2) customer-centric analysis; 3) policy-centric analysis; 4) process-centric analysis; and 5) fraud-centric analysis [16].

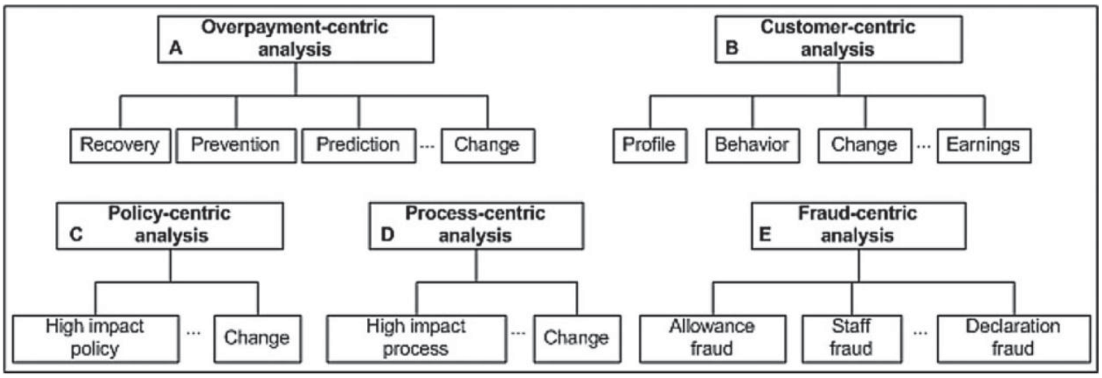


Figure 3-2 SSDM Goals

They are shown in Fig. 3.2 and are explained briefly in the following.

1) *Payment-Centric Analysis:* Overpayments or government customer debt are a major concern in social security government services [119]. Overpayment/debt-centric analysis, therefore, emerges as a major objective of SSDM. Its goals consist of the deep understanding of the distributions of overpayments across business lines, the cause and effect of overpayments, and the evolution and changes of overpayments in the life of government customers. In addition, issues that are related to payment accuracy also involve underpayment analysis, and alignment and gap analysis between customer earning/employer payment and government payment. The findings

from payment-centric analysis contribute to government customer debt recovery, debt prevention, and debt prediction, as well as better customer service quality.

2) *Customer-Centric Analysis:* Customer-centric analysis in SSDM aims to deeply understand which customers cause overpayments, and the reasons and indicators behind those customers who owe the government [119]. The reasons may be related to customer profiles, behaviors, earnings, and so on, as well as changes to any of these aspects. The findings from customer-centric analysis contribute to evidence, indicators, and observations that assist the understanding of why, when, and how some customers cause overpayments when others do not. In addition, customer risk rating and customer service recommendations are other objectives.
3) *Policy-Centric Analysis:* Policy-centric analysis in SSDM seeks to deeply understand which policies [121] are associated with overpayments, and the reasons and indicators behind these policies. Other analysis may focus on the relationships between policy changes, overpayments, and customers. Identifying those policies and policy changes that have led to, or are associated with, overpayments could be used to prevent the occurrence of debts and to actively manage customers.
4) *Process-Centric Analysis:* Process-centric analysis in SSDM is carried out to deeply understand what business processes or process changes are associated with overpayments [119], [121], as well as the reasons and indicators behind them. By analyzing the relationships between processes, overpayments, and customers, social security government officers obtain a deep understanding of what could be optimized in business processes or during process changes in order to minimize overpayments or the probability of debt occurrence.
5) *Fraud-Centric Analysis:* Fraud-centric analysis in SSDM is undertaken to analyze whether fraud takes place in the social security business, and where, why, and how fraud happens and evolves [133]. Analysis can be conducted on child welfare fraud, allowance fraud, declaration fraud and staff fraud, and the resultant findings that are used to assist the detection, prevention, and prediction of fraud in the social security business.

3.3 Social Security Data Mining Challenges

The data mining tasks listed are comprehensive, involving many aspects of both traditional and emerging data mining techniques. Some can be handled by the utilization of existing general data mining techniques, while others have to be dealt with using revised or newly developed methodology and approaches in order to handle the mixture of social security business and general data complexities. On the basis of our observations of the challenges involved in conducting the aforementioned data mining tasks, we discuss the following key challenges (see Fig. 3.3) in terms of the main procedures of social security data processing, pattern analysis, and knowledge delivery.

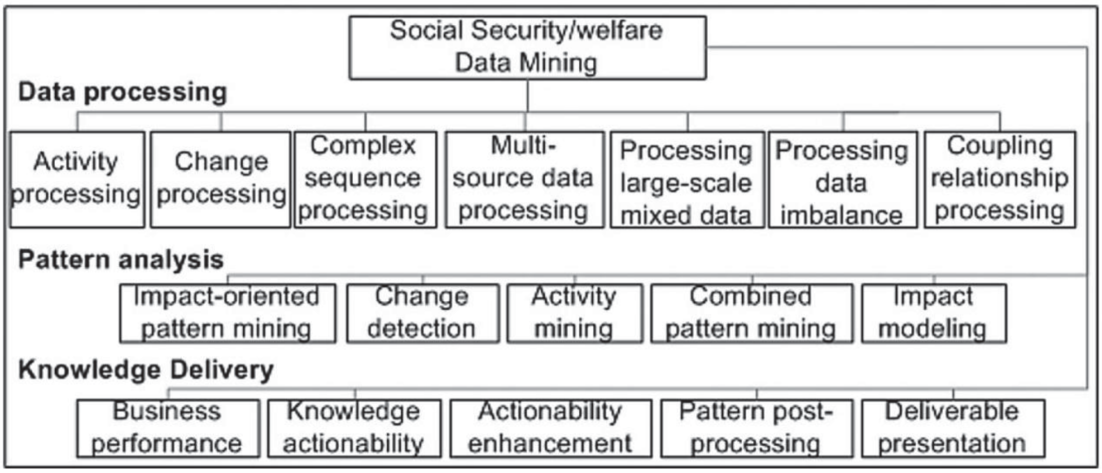


Figure 3-3 SSDM Challenges

3.3.1 Social Security Data Processing

The processing of data characteristics in the social security business shares many similar data processing issues within the data mining and machine-learning community. In particular, the following areas [16] are especially, important in SSDM.

1) *Activity processing:* The processing of activities and activity sequences [24] needs to address complex features, such as temporal, spatial, structural and semantic dimensions, as well as handle issues such as data sparsity and imbalance [23], [107], dynamics, and associated impact [25] on business (such as causing overpayments) in activity feature selection, activity extraction, activity sequence construction, and preparation.

2) *Change processing:* Changes are widely dispersed in social security data [119], [131]. The consideration of change data in SSDM is crucial to identify meaningful causes and patterns [19]. Change data processing involves issues such as change definition, representation of change, interactions and relations between changes and other entities, and change feature extraction.
3) *Complex sequence processing:* Customer–government interaction generates intensive activities in the social security business. Sequences are complicated [22]–[24] if the relevant information is considered; for instance, the time an activity takes place and the reason (it could be another activity) for the activity occurrence. For family based debt investigation, it is probably necessary to put all family members' activity sequences together to observe the differences, which will involve multiple coupled sequences [19]. The processing of such sequences involves issues such as representation of single and multiple coupled sequences, modeling sequence relations, sequence feature extraction, and data structure design for storing sequences, and relevant information.
4) *Multisource data processing:* Very often, SSDM has to engage multiple sources of social security data, since more informative patterns reflecting the actual business picture can only be identified on multisource data [22], [110]. Meaningful SSDM analysis involves data of customer-officer interaction transactions, customer demographics, government policies, business processes, customer registration data, customer earnings, debt outcomes, and debt recovery arrangements [121]. It is necessary to correctly understand the relationship between different sources of data from business logic, syntactic and semantic aspects, how to align and fuse them (for instance, whether from the data or pattern [22], [112] perspective) while considering the intrinsic business logic, and how to deal with different granularities.
5) *Processing large-scale mixed data:* SSDM tasks often involve large-scale mixed data. For instance, a debt usually occurs and exists for several months to a few years [130], and the investigation of debt drivers needs to involve multisource information recorded in different structures and formats [121]. Among other things, this requires determining the timeline to select and align different sources of data [22], [110], a smart data structure to fit relevant information, proper sampling methods, an efficient

strategy to scan/filter the data, and the selection and fusion of mixed features in both processing and pattern mining [113].

6) *Processing data imbalance:* Besides the normal techniques available in the literature about class imbalance [23], [107], such as under sampling and oversampling, it is expected that greater effort will be necessary in processing extremely imbalanced data in a large scale set and on designing proper data structures, filtering, and sampling algorithms to prepare both class and item-set imbalanced data.
7) *Coupling relationship:* The representation of coupling relationships [18], [19], first involves the definition and extraction of coupling within an entity and between entities from syntactic and semantic perspectives. Further work concerns how to manage the relationships and store the relevant data by developing a suitable data structure, a representation system, and data extraction mechanism.

3.3.2 Social Security Pattern Analysis

While many traditional and emerging pattern mining methods and pattern types can be applied directly to SSDM, we also observe specific needs emerging from mining social security data [16]. These are briefly discussed below, and the observations can certainly be used for mining other, similar applications.

1) *Change detection:* Challenging issues in detecting changes [131] cover many possibilities, e.g., representing changes and change contexts in organization–customer interactions, tackling data complexities in processing and mining change-centered data, identifying change patterns in customer circumstance and behavior contexts, identifying group relationships and group behavior changes, identifying customer interaction changes in response to policy/procedure changes, adapting the detection of customer and group dynamics, and extracting and evaluating noncompliance drivers based on the mined change patterns.
2) *Activity mining:* Activities [23], [24] are widely seen in social security data, which create a challenge for traditional data mining [24]. Mining activity patterns can focus on activity-centric, impact-centric, or customer-centric analysis [24], and each aspect is new. In addition, the evaluation of activity mining is nonexistent, and therefore, new interestingness metrics need to be developed for each activity mining method.

3) *Impact modeling* measures the impact of certain data on business [23], [116]. The nature of the impact, how to measure it and how it is associated with patterns and debt occurrence, is open to investigation. The measure of impact needs to be specified in terms of target data and customer groups by involving domain knowledge and needs to be evaluated by domain experts.

4) *Impact-oriented pattern mining* identifies patterns that are associated with specific impact. Unlike traditional patterns that consist of items only, impact-oriented patterns have two facets: one is item sets, the other is the impact associated with the item sets. As discussed in [23], impact-oriented pattern mining is challenging, since many emerging pattern types may be identified, such as positive impact-oriented patterns, negative-impact-oriented patterns, impact-contrasted patterns, and impact-reversed patterns [22], [23].

5) *Combined mining* [22], [105], [110], [112] is a two to multistep data mining procedure, consisting of mining atomic patterns, merging atomic pattern sets into a combined pattern set, or merging dataset-specific combined patterns into the higher level of a combined pattern set if there are multiple datasets. Combined mining is essential in SSDM, which mines for combined patterns by engaging multisource data and complicated social security data, as discussed in the previous section on data processing. Mining combined patterns is not easy and involves the invention of new techniques and methods. For instance, the authors in [26] have discussed new methodologies, including multifeature combined mining, multimethod combined mining, and multisource combined mining [22]. Combined mining can lead to creative pattern types, such as pair pattern, cluster pattern, incremental pair pattern, and incremental cluster pattern [22], in which pattern components are coupled in terms of relationships such as peer-to-peer or master–slave.

3.3.3 Knowledge Delivery

In actionable knowledge discovery [17], [20], [26], it is important to deliver knowledge of business interest which can be taken over by business people. This is not a trivial problem, as discussed in domain-driven data mining [7], and is applicable to SSDM.

1) *Business performance* is the performance of patterns from the business perspective [17]. While data miners usually concentrate on the technical performance evaluation of patterns, the specification and evaluation of business performance will certainly provide additional information for business people to judge the value of the findings. How to define and measure business performance from subjective and objective perspectives is worthy of research, as well as which business metrics need to be defined for use to generally measure business impacts associated with patterns [20].
2) *Knowledge actionability* is the actionability of identified patterns. The authors in [17] propose a general framework of knowledge actionability and highlight the engagement of both technical and business performance from subjective and objective perspectives in measuring knowledge actionability. In the social security area, our job is to specify metrics to measure SSDM knowledge actionability.
3) *Actionability enhancement* [17], [20] concerns enhancing the actionability of identified patterns. While many aspects can be addressed [17], it is not a straightforward task. It is worthwhile to analyze why the discovered knowledge is not actionable, what aspects can be focused on, and what actions can be taken to enhance the actionability.
4) *Pattern post processing* is an important way to enhance knowledge actionability and is applicable to SSDM. The authors in [113] summarize the main techniques to be developed or enhanced in postprocessing and postmining and collect the latest work on post mining of association rules. Considering the social security data specialization, new postprocessing techniques need to be developed, with the involvement of domain knowledge.
5) *Deliverable representation* [17] builds appropriate mechanisms to convert SSDM findings into business-oriented deliverables and represents deliverables in a business friendly manner. This is actually a challenging issue, which has not been well studied. Because of the engagement of customer–government interaction, in particular, the relevant deliverables need to show the interactive procedures by reflecting the underlying activity patterns. In the following chapters, we introduce a number of case studies of SSDM that address some of the aforementioned challenges.

3.4 Social Security Data Mining Tasks

To support the data mining goals that are discussed above, many tasks need to be performed in SSDM. Traditional data mining methods, including association rule mining, frequent pattern mining, clustering, and classification, will certainly play an important role in achieving the aforementioned goals. We categorize the SSDM tasks [16] as follows, according to the main entities in social security/welfare business [121] and to address the SSDM goals, as shown in Fig. 3.4: 1) data processing; 2) activity analysis; 3) customer risk analysis; 4) earnings analysis; 5) change detection; 6) debt analysis; and 7) fraud detection.

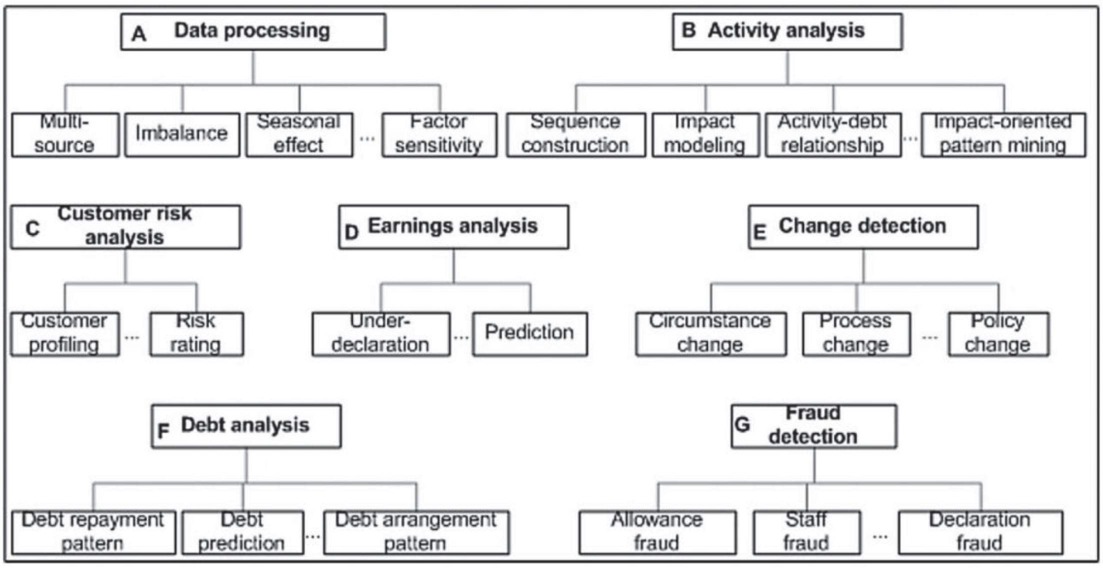


Figure 3-4 SSDM Tasks

We briefly explain these tasks in the following sections.

3.4.1 Data Processing

The main tasks in social security data processing have many aspects, including many common data processing issues discussed in the community. In particular, we have the following.

1) SSDM involves *multiple data sources*: for instance, customer demographic data, customer interaction transactions with government officers, arrangement and repayment activity data, and debt-related data. Therefore, it is essential to deal with multiple sources of data [22].

2) *Imbalance:* Overpayment-related data only consist of a very small proportion of the whole social security data. Debt-related pattern analysis has to identify patterns in imbalanced datasets [23].

3) *Seasonal effect:* Social security government services present very strong seasonal characteristics that are determined by service objectives and policies. For instance, during holiday seasons, many immigrants move to their mother country, taking children to visit relatives. This may lead to gaps in reporting, although the government may continue to pay the usual rate, resulting in a government overpayment [130].

4) *Factor sensitivity:* This reflects the fact that not all variables and values contribute equally; some variables may play only a small role or duplicate others. Factor impact analysis, principal component analysis, and feature mining may be necessary in analyzing the sensitivity and interrelationships amongst factors, variables, and features.

3.4.2 Activity Analysis

Activity data [24], [120] refer to the interactive events, operations, and actions occurring in social security business. They form the main component of behavioral data [15], and the analysis of activities is complicated [24]. We discuss several tasks here.

1) *Activity sequence construction:* This involves activity types, activity distribution, activity relationships, timeline, and so on. The exploration of these aspects can generate useful hints for the construction of activity sequences [23], [120].

2) *Activity impact modeling:* In business, each activity or activity series plays a different role, and some activities contribute more than others. Different combinations of activity sequences may lead to a variety of outcomes. Before constructing activity sequences, there is a need to understand and quantify the outcomes and the impact of activities associated with a particular business, e.g., debt occurrence. Measures and models need to be developed to specify and differentiate the impact of particular activities [23], [24].

3) *Activity–debt relationship analysis:* In the social security domain, the occurrence of debt is largely driven by activities or activity sequences. The analysis of relationships

between activities and debt [23], [24] aims to determine which activities are more sensitive to debt occurrence, how they result in debt (e.g., as a group, or before or after the debt occurrence), and to what extent an activity (sequence) leads to debt.

4) *Impact-oriented pattern analysis:* While general activity patterns can be identified, business people are more interested in those activities that are associated with high business impacts, which we call *impact-oriented patterns* [23], [24]. Mining high business impact-oriented patterns is not easy, since it may involve the handling of activity imbalance, impact definition, complex pattern types, and the definition of new interestingness metrics.

3.4.3 Customer Risk Analysis

As a result of having a deep understanding of customers, any customer or customer group can be ranked in terms of the risk of causing overpayments. To rate customers requires consideration of various scenarios and risk specifications.

a) *Customer profiling [130]:* This creates a comprehensive understanding of which customer profiles lead to debt at different probability levels, and which profile-based factors are more sensitive to which allowance-based debt occurrence. Customer profiling needs to be more deeply conducted by scrutinizing customer circumstances, distributions, structures, relationships, and their variations. It is worthwhile to analyze the relationships between these aspects and debt occurrence and debt impact.

b) *Customer risk rating [126]:* While it is known that some customers are more likely to be associated with debt occurrence than others, it is advantageous to specify their particular risk and risk rate. This is associated with issues, such as risk types, which customer-related factors contribute to risk from the perspective of demographics, behaviors and change, and time sensitivity to risk occurrences.

It is also interesting to see whether some customers have a higher probability than others of causing debt before they register an allowance, and the key factors causing such a difference. Information declared by a customer to the government certainly affects the likelihood of risk and debt. We are interested in the relationship between the information declaration level and coverage and the risk level.

3.4.4 Earnings Analysis

Incomes and earnings are particularly sensitive to social security debt and the delivery of service objectives. Any deliberate manipulation or unintentional disregard of earnings could lead to eventual overpayments. Therefore, earnings analysis not only concerns the relationship between what has been declared and the debt occurrence, but also what has been under-declared or is missing.

a) *Manipulative declaration analysis:* The manipulation of earnings declaration has been viewed as one of the key drivers of debt [127], [132]. Earnings may be under-declared or neglected in reporting to the government, and the direct detection of manipulative declaration is often difficult, since it is not easy to identify the evidence. The identification of manipulative declaration needs to capture what a customer reports to the government at the initial registration, customer behavioral data, customer circumstance changes, family-based behaviors and situation changes, and customer activities related to expenses, which is often a very complex issue.

b) *Earnings prediction [132]:* While it is difficult to achieve, the prediction of earnings for correctly and manipulatively declared customers who are eligible for government benefits can assist with the early detection, and thus prevention, of debt. The detection of earnings for correctly declared customers is generally more manageable than for manipulators. Besides numerical data-based prediction techniques, new prediction methods are essential by combinatorially considering customer circumstance changes, behaviors, membership data and changes, etc.

3.4.5 Change Detection

In organization–customer interactions, significant changes, occurring either in customer circumstances and behaviors, or in business policies and processes, may lead to noncompliance, resulting in inconsistencies or even substantial financial losses and damaging effects on an organization [131]. For instance, changes in customer demographics may not be instantly reflected in relevant business lines, thereby resulting in inconsistencies or overpayments. An example is the almost $2 billion Centrelink customer debt, which the 2007-2008 ANAO audit report [119] concludes that customer debt arose primarily from customers failing to notify Centrelink of changes in circumstances. This is a challenging issue.

a) *Customer circumstance change analysis:* Customers often experience change in their circumstances, e.g., changing their home address or educational status. Any significant circumstance change could lead to debt, and the detection and early prediction of such significant circumstance changes are critical for managing debt. In addition, changes that are associated with specific customer groups are interesting. Identifying such groups and their behavior and circumstance changes is useful in understanding the reasons for debt occurrence.
b) *Policy change analysis:* Some policy changes have been shown to be related to debt occurrence. A policy change may affect certain customers' behaviors and declarations and eventually lead to more debt. It is *worthwhile* to analyze the relationships between particular policies (or policy groups), customer behavior changes, declaration changes, and debt occurrence. Key issues include the analysis of which policy changes are most likely to cause debt, and why that happens. The findings can then be used to advise policy changes and to take steps toward customer intervention for certain policy changes.
c) *Process change analysis:* Similar to policy changes, some process changes are more sensitive to debt increases than others. The analysis of relationships between process changes and debt changes can alert process makers to intervene in certain process changes to prevent associated debt. Similarly, it is helpful to combinatorially analyze process changes, customer behavior changes, declaration changes, and debt changes.

3.4.6 Payment Accuracy Analysis

As the actual outcome of unfair and wrongly directed social security delivery, the payment and debt-centric analysis is designed to discover direct characteristics, distributions, causes, and changes associated with debt occurrence. Many aspects can be studied in payment and debt analysis; for instance, debt statistics to generate the distribution of debt and the dynamics of debt development.

a) *Debt recovery pattern analysis* [103], [115] is conducted to analyze patterns of debt-recovery-related activities. This can be conducted on the recovery activity sequences, time interval distributions for different recoverable groups, recovery speed analysis, recoverable customer characteristics, unrecoverable customer circumstance patterns, effectiveness of recovery methods on different customer groups, and early recovery recommendation.

b) *Debt arrangement pattern analysis* [23] analyzes patterns of government arrangement-related activities and methods. It is worthwhile to analyze the arrangement effectiveness of arrangement methods and intervals in relation to different customers or customer groups. With the findings of this research, more pertinent arrangement methods and time arrangements can be made for particular groups.

c) *Debt repayment pattern analysis* [23] is carried out to analyze patterns of debt-repayment-related activities and repayment groups. This can be through customer repayment activity sequences, repayment time interval distributions, repayment method distribution, fast–slow repayment characteristics, effectiveness of different repayment methods on different groups, and effective repayment recommendations to particular groups.

d) *Debt arrangement–repayment analysis* [23]: By combining the analysis of debt recovery arrangements and customer repayments, more actionable patterns can be identified to advise the effective arrangement–repayment combinations for particular customer groups to enable more effective recovery. The combinatorial analysis of customer circumstances, arrangement methods, repayment methods, intervals, debt recovery speed, etc., can lead to very informative intervention rules for debt recovery.

e) *Debt prediction* [103], [108], [115] aims to predict which customers or customer groups will incur a debt, on which benefit services, and when. By focusing on particular customers or customer groups, the task is to predict when, or at what time interval, a debt will occur, the likely frequency of debt occurrence, the likely size of the debt, and so on.

f) *Driver analysis* is for customers with either overpayment or underpayment, or for those with misaligned payment between earnings and entitlement. Detection of drivers may be conducted from many aspects, such as the differences between over- and underpayment groups, changes associated with normal and incorrect payment groups, and behavioral patterns associated with earnings declaration of those customers whose income is misaligned with the entitlement.

3.4.7 Fraud Detection

Fraud may take place in many aspects of the business.

a) *Allowance fraud* [121], [133] is that which takes place on allowances. There are many types of allowances that are available for eligible customers from government services. A customer cheating the government to obtain an allowance to which they are not entitled causes an allowance–customer mismatch fraud. In other cases, an eligible customer may cheat the government in order to maximize the amount of payment, resulting in overclaim allowance fraud. Allowance fraud detection aims to detect the corresponding key factors contributing to allowance-customer mismatch frauds and overclaim allowance fraud, and the patterns and reasons that relate to them.

b) *Declaration fraud* [121], [133] is the fraud that takes place on declarations, which may be manipulated by fraudsters. Declaration fraud detection identifies factors, scenarios, patterns, and changes of underdeclaration, delayed declaration, and missing declaration in terms of allowance types and customer groups. Different allowance types and customer groups may experience a variety of declaration patterns. Other work includes the prediction of manipulative declarations.

c) *Staff fraud* [121], [133] is the acquisition of payments by an employee, for himself or herself or for another person, through dishonesty or deception. A staff member may create false documents and process false benefit claims against genuine customer records. Staff fraud detection seeks to identify key factors, business sections, methods, and patterns related to the fraudulent behavior of staff and the impact of such behavior. It is often difficult to identify staff fraud because it involves complex data, business process, and lack of evidence.

4 Positive/Negative Social Security Sequential Rules Analysis

4.1 Positive/Negative Sequential Rules

4.1.1 Introduction

Whereas traditional association rule mining [151] and sequential pattern mining [152] deal only with the positive relationship between itemsets or events, it might be interesting to examine frequent sequential patterns that display a 'non-occurrence' of some items – which is referred to as *negative sequential patterns*.

Although there is a considerable amount of literature dealing with negative association rules designed to find such kind of relationships [153,154,155,156], the application of such rules to discover negative sequential patterns is seldom mentioned.

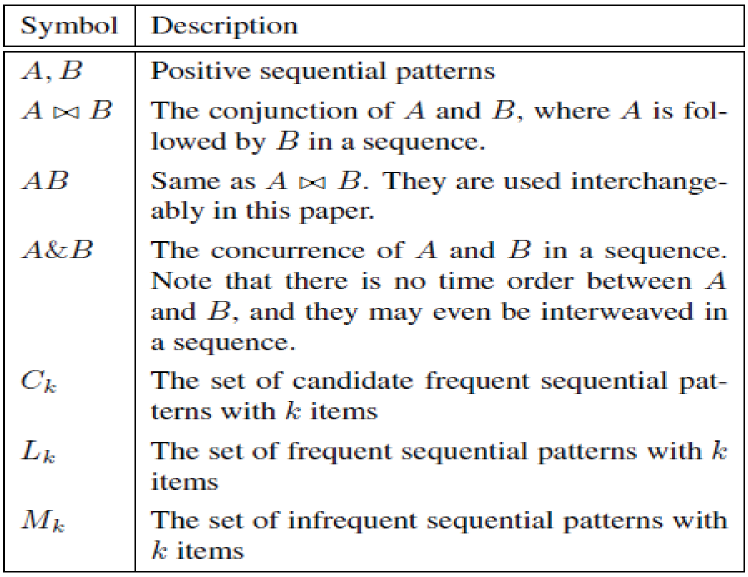
The following sections present an analysis of three types of negative sequential rules as well as a new technique to find event-oriented negative sequential rules which involves a study of the negative relationship in sequence data defined in two forms:

- *negative sequential patterns (NSP)* like $a \neg(bc)da \neg e$, where a to e are events/items; and
- *negative sequential rules (NSR)* like $A \rightarrow \neg B$, $\neg A \rightarrow B$ and $\neg A \rightarrow \neg B$, where A and B are positive sequential patterns composed of items in time order.

From NSP we can derive sequential rules such as $a \neg(bc)d \rightarrow a \neg e$, which we name *generalized negative sequential rules (GNSR)*. There are some preliminary works on negative sequential pattern mining [157,158,159,160], but collectively it was inefficient for mining such patterns.

To solve the problem the idea of *negative sequential rules* is developed – whose left and/or right sides could be negations of traditional positive sequential patterns; and came up with a new efficient algorithm designed for mining *event-oriented negative sequential rules* – where the right side of a rule is a single event or its negation. The notations are provided in Table 4.1.

Table 4-1 Notations



Symbol	Description
A, B	Positive sequential patterns
$A \bowtie B$	The conjunction of A and B, where A is followed by B in a sequence.
AB	Same as $A \bowtie B$. They are used interchangeably in this paper.
$A\&B$	The concurrence of A and B in a sequence. Note that there is no time order between A and B, and they may even be interweaved in a sequence.
C_k	The set of candidate frequent sequential patterns with k items
L_k	The set of frequent sequential patterns with k items
M_k	The set of infrequent sequential patterns with k items

4.1.2 Background and Related Work

Techniques concerning negative association rules focus principally on finding rules in the form of $A \to \neg B$, $\neg A \to B$ and $\neg A \to \neg B$, and these are referred to as either *confined negative association rules* [156,154], or *generalized negative association rules* – the latter containing a negation of an item, such as $A \wedge \neg B \wedge \neg C \wedge D \to E \wedge \neg F$ [155].

Whereas the idea of sequential pattern mining is to find frequent patterns in data such as transactional data [152], the difference between this and frequent itemset mining is that ‘time order’ is taken into account in sequential pattern mining. Some well-known algorithms for sequential pattern mining include AprioriALL [152], GSP (Generalized Sequential Patterns) [161], FreeSpan [165], PrefixSpan [162], SPADE [163] and SPAM (Sequential Pattern Mining) [164].

With regard to sequential patterns, the non-occurrence of an element may also be interesting. For example, in the social welfare environment on which this thesis is based, failing to update a customer record after a change of address has been advised may result in an overpayment to that customer and a subsequent debt. Such sequences, that is, those with the non-occurrence of elements, are referred to as *negative sequential patterns*. And while most research on sequential patterns is focused on positives, there is some that looks at negative sequential patterns. For example:

- Sun et al [160] proposed negative event-oriented patterns – a negative rule in the form of $\neg P\ T\rightarrow e$, where e is a target event, P is a negative event-oriented pattern, and the occurrence of P is unexpected rear in T-sized intervals before target events. It is a special case of negative sequential pattern $\neg A \rightarrow B$.

  In their work, P is regarded as an 'existence pattern' (a frequent itemset without time order) instead of a sequential pattern;
- Bannai et al [157] proposed a method for finding optimal pairs of string patterns to discriminate between two sets of strings. The pairs are in the forms of $p' \wedge q'$ and $p' \vee q'$, 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. The substring can be taken as a special case of sequential pattern, where the elements in the patterns are continuous;
- Ouyang and Huang [159] proposed negative sequences as $(A, \neg B)$, $(\neg A,B)$ and $(\neg A, \neg B)$. Their idea is generating frequent itemsets first, based on which both frequent and infrequent sequences are found, and then negative sequential patterns are derived from infrequent sequences. A drawback of their algorithm 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 [158] 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; and
- Chen et al [158] designed a technique PNSP (Positive and Negative Sequential Patterns Mining) for mining positive and negative sequential patterns in the form of $(abc\ \neg(de)(ijk)$. They proposed some constraints for negative sequential patterns. For

example, valid negative sequential patterns do not contain contiguous absence of elements and an itemset must be frequent to make it a valid negative itemset. Their method is broken into three stages :

1) all positive sequential patterns are found
2) all positive itemsets are found, from which all negative itemsets are derived and
3) both positive and negative itemsets are joined to generate candidate negative sequential patterns, which are in turn joined iteratively to generate longer negative sequences in an Apriori-like way.

4.1.3 Problem Statement

4.1.3.1 Negative Sequential Rules

The negative relationships in transactional data are defined as follows.

Definition 1 (Negative Sequential Rules (NSR)). *A negative sequential* rule is in the form of A →¬B, ¬A→B or ¬A→¬B, where A and B are positive sequential patterns composed of items in time order.

Definition 2 (Event-oriented Negative Sequential Rules (ENSR)). *An* event-oriented negative sequential rule is a special NSR, where the right side B is a single event, that is, the length of B is one.

Definition 3 (Impact-oriented Negative Sequential Rules (INSR)). *An* impact-oriented negative sequential rule is a special ENSR, where the right side is a predefined target outcome T, such as a specific class or a predetermined event.

Definition 4 (Negative Sequential Patterns (NSP)). *A negative sequential* pattern is a sequence of the occurrence or non-occurrence of items in time order, with at least one negation in it.

Definition 5 (Generalized Negative Sequential Rules (GNSR)). *A generalized* negative sequential rule is in the form of A → B, where one or both of A and B are negative sequential patterns.

Based on the above definitions, we can get: $I_{GNSR} \supset I_{NSR} \supset I_{ENSR} \supset I_{INSR}$, where I_{GNSR}, I_{NSR}, I_{ENSR} and I_{INSR} denotes respectively the sets of the above four kinds of rules. Although an INSR looks similar to an ENSR, the former is specially focused on a specific subset of ENSRs, so it demands more efficient techniques tailored for its special needs.

4.1.3.2 Types of Negative Sequential Rules

Traditional sequential rules are positive sequential rules, which are in the form of $A \rightarrow B$, where both A and B are positive sequential patterns. It means that pattern A is followed by pattern B. We refer to such positive rules as Type I sequential rules. By changing A or/and B to its/their negations, we can get the following three types of negative sequential rules:

– 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 doesn't appear, then pattern B will not occur.

For types III and IV whose left sides are the negations of sequences, the meaning of the rules is: if A doesn't occur in a sequence, then B will (type III) or will not (type IV) occur in the sequence. That is to say, there is no time order between the left side and the right side. Note that A and B themselves are sequential patterns, which makes them different from negative association rules. However, if time constraint is considered in sequential rules, the last two types of rules may have new meanings, which is out of the scope of this paper. The supports, confidences and lifts of the above four types of sequential rules are shown in Table 4.2. 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.

Table 4-2 Supports, Confidences and Lifts of Four Types of Sequential Rules

Type	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	$\neg A \rightarrow \neg B$	$1-P(A)-P(B)+P(A\&B)$	$\frac{1-P(A)-P(B)+P(A\&B)}{1-P(A)}$	$\frac{1-P(A)-P(B)+P(A\&B)}{(1-P(A))(1-P(B))}$

4.2 Efficient Mining of Event-oriented Negative Sequential Rules

A *negative sequential rule (NSR)* is defined as a rule in the form of $A \rightarrow B$, where either or both sides of the rule is/are the negations of positive sequential patterns. A negative sequential rule with the right side as a single event is referred to as an *event-target negative sequential rule*. Therefore, the target problem is as follows:

Given a database of sequences and three user-specified parameters, minimum support *minsupp*, minimum confidence *minconf* and minimum lift *minlift*, find all event oriented negative sequential rules whose support, confidence and lift are no less than *minsupp*, *minconf* and *minlift*, respectively. In this work, we consider the case where every transaction is composed of one item only. That is, the sequences look like *abcad*, instead of *a{bc}ad*, where *b* and *c* occur in one transaction in the latter case.

4.2.1 Algorithm for Mining Event-Oriented Negative Sequential Rules

To discover event-oriented negative sequential rules, SPAM (Sequential Pattern Mining) [164] is used as a starting point. Although SPADE and PrefixSpan were also available, SPAM is chosen as it uses a vertical bitmap representation of the database and a depth-first search strategy for efficient mining of sequential patterns. It also searches the sequence lattice in a depth-first way, and candidates of longer sequences S_g are generated by appending frequent items *{i}* to existing frequent sequences S_a.

The SPAM algorithm is very efficient in that it uses bitmap to count the frequency of sequences and only the nodes in the path to the current node are kept. The candidate generation of SPAM is composed of two steps: *S-step* and *I-step*. The *S-step* appends *i* to S_a, which builds a longer sequence $S_g = S_a \bowtie i$. The *I-step* adds *i* to the last itemset of S_a, which builds a new sequence of the same length as S_a. Since we consider transactions with only one item, an element in a sequence is a single item instead of an itemset. Therefore, only *S-step* from SPAM is used here.

The frequency of items (L_1 *sequence*) is counted, which is subsequently used for the computation of sequential rules. Since the algorithm works with a depth-first search strategy, the support of S_a was available in the parent node, while the support of item *i* was available at the very beginning. Because the bitmaps of S_a and *i* were available at each node for counting the support of $S_a \bowtie i$ (with a bit-wise AND-operation), the support of $S_a \& i$ was easily derived.

Figure 4.1 on the following page gives the pseudo code for finding negative sequential rules, which is based on the function Find Sequential Patterns() from SPAM. A recursive call goes down the search lattice to find sequential rules. For each possible extension *i* at a level, the support is computed with a bit-wise AND of post-processed bitmap with the candidate frequent-1 itemset (see lines 2-6). If the result shows that the support is no less than a given support threshold, then *i* is added to the extension list of the node at next level and the confidence and lift for positive sequential patterns (Type I) is calculated (see lines 8-14). Otherwise, the support, confidence and lift for negative sequential patterns of Types II, III and IV is computed (see lines 16-33). If the support, confidence and lift are all above predefined thresholds, the corresponding rules are outputted. Then the node at next level is checked recursively to find all sequential rules (see line 35). At each level of the sequence lattice, only one node is kept in memory, which makes it very space efficient.

ALGORITHM: FindNegativeSequentialRules, a recursive call that goes down the search lattice to find negative sequential rules
INPUT: *curNode*: information about the current node
OUTPUT: event-oriented negative sequential rules

```
FOR each possible s-extension $i$ from this level
    /* AND the post-processed s-step bitmap with the candidate frequent-1 itemset; */
    $cntA$ = curNode.parent.count;
    $cntB$ = $i$.count;
    $tempAndBitmap$=Bit-Wise-And(post-processed bitmap of curNode, bitmap of $i$);
    $cntAB$=$tempAndBitmap$.count; /* corresponding to $P(AB)$ */
    IF $cntAB \geq minsupp * totalCust$     /* $totalCust$ is the number of customers/sequences. */
        add $i$ to nextNode's s-extension list;
        /* generate positive sequential patterns */
        // Type I: $A \rightarrow B$
        $supp = cntAB/totalCust$;
        $conf = cntAB/cntA$;
        $lift = cntAB * totalCust/(cntA * cntB)$;
        output "$A \rightarrow B$" if $supp \geq minsupp$, $conf \geq minconf$ and $lift \geq minlift$;
    ELSE
        /* generate negative sequential patterns */
        $tempOrBitmap$=Bit-Wise-Or(bitmap of curNode, bitmap of $i$);
        $cntAorB$ = the count of "1" in $tempOrBitmap$;
        $cntAandB = cntA + cntB - cntAorB$; /* corresponding to $P(A\&B)$ */
        IF $P(A\&B) < P(A)P(B)$          /* $P(AB) \leq P(A\&B) < P(A)P(B)$ */
            compute support, confidence and lift according to Equations 5, 7 and 8.
            output "$A \rightarrow \neg B$" if $supp \geq minsupp$, $conf \geq minconf$ and $lift \geq minlift$;
            compute support, confidence and lift according to Equations 10, 12 and 13.
            output "$\neg A \rightarrow B$" if $supp \geq minsupp$, $conf \geq minconf$ and $lift \geq minlift$;
        ELSE IF $P(AB) < P(A)P(B)$       /* $P(AB) < P(A)P(B) \leq P(A\&B)$ */
            compute support, confidence and lift according to Equations 5, 7 and 8.
            output "$A \rightarrow \neg B$" if $supp \geq minsupp$, $conf \geq minconf$ and $lift \geq minlift$;
            compute support, confidence and lift according to Equations 15, 17 and 18.
            output "$\neg A \rightarrow \neg B$" if $supp \geq minsupp$, $conf \geq minconf$ and $lift \geq minlift$;
        ELSE                              /* $P(A)P(B) \leq P(AB) \leq P(A\&B)$ */
            compute support, confidence and lift according to Equations 15, 17 and 18.
            output "$\neg A \rightarrow \neg B$" if $supp \geq minsupp$, $conf \geq minconf$ and $lift \geq minlift$;
        END IF
    END IF
    FindNegativeSequentialRules(nextNode); /* A recursive call to check the node at the next level */
END FOR
```

Figure 4-1 Algorithm for Discovering Event-oriented Negative Sequential Rules

4.2.2 Experimental Evaluation

The algorithm (which I refer to as *SpamNeg*) is implemented with C++ based on the source code of the SPAM algorithm from http://himalayatools. sourceforge.net/Spam/ and tested it with a synthetic dataset generated with IBM AssocGen Transactional Data Generator. All tests were conducted on a PC with Intel Core 2 CPU of 1.86GHz, 2GB memory and Windows XP Professional SP2. The number of items per transaction was set to one when generating datasets. We first tested the algorithm on a dataset with 10,000 customers with 50 items per sequence and the length of maximal patterns as 20.

The result of setting the minimum supports ranging from 0.2 to 0.7 is shown in Figure 4.2, which shows that both *SpamNeg* and SPAM run faster with larger minimum support because the search space becomes smaller. Moreover, *SpamNeg* needs longer running time than SPAM and the extra time is caused by generating negative sequential rules when the support of a candidate is less than the support threshold. The scalability of the algorithm was tested on datasets with average sequence length as 40, length of maximal patterns as 10. The number of customers ranged from 10,000 to 60,000, and the support threshold was set to 0.3.

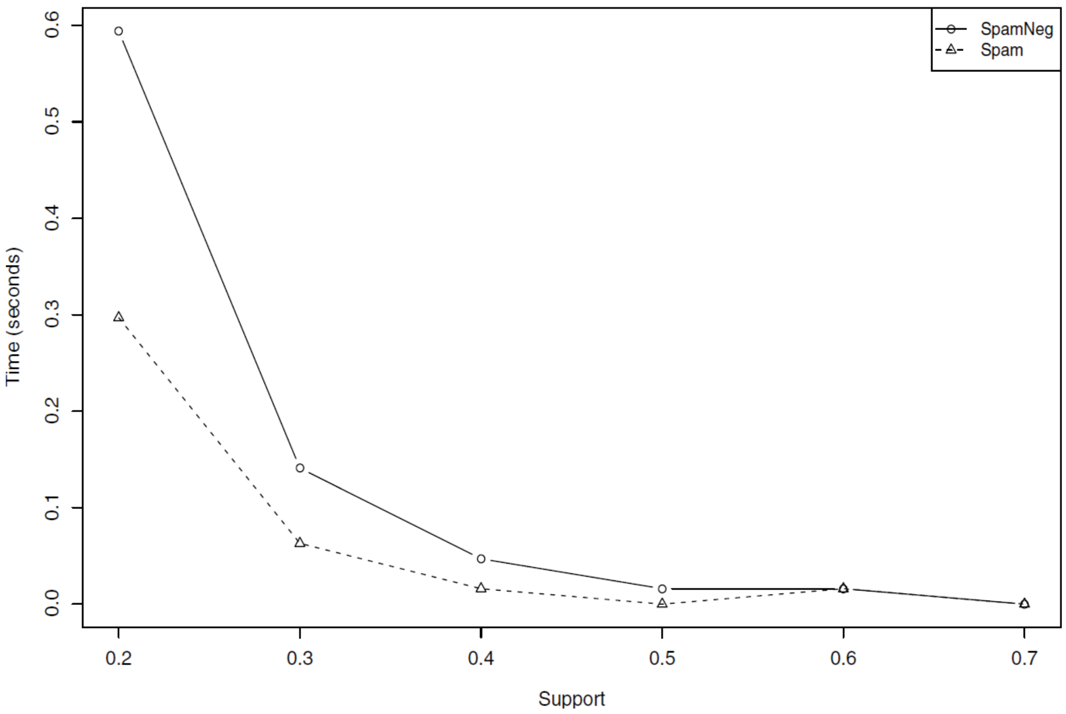


Figure 4-2 Scalability with Minimum Support

Figure 4.3 shows the result of the above test. It is clear from the figure that *SpamNeg* is linear with the number of sequences. The running time with varying sequence lengths is also shown.

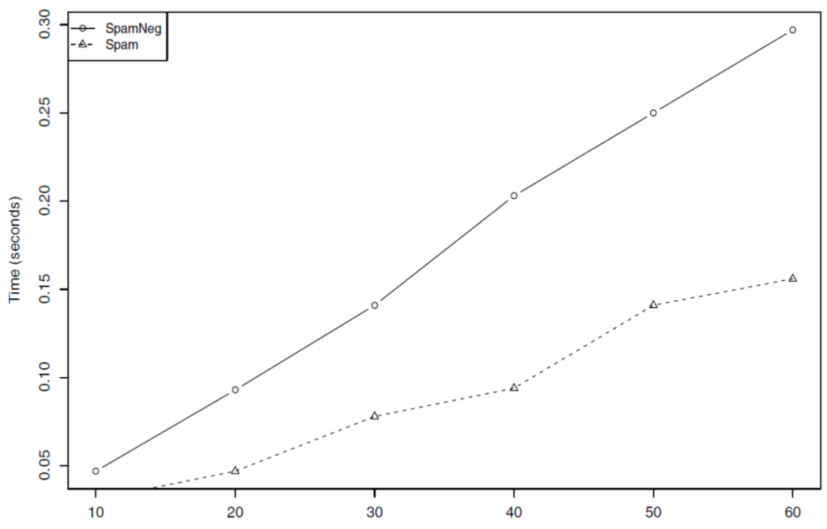


Figure 4-3 Scalability with the Number of Sequences

With a dataset of 100,000 customers, the length of maximal patterns as 20, and the average sequence length ranging from 10 to 45, the support threshold is set to 0.2. Fig. 4.4 shows that the running time becomes longer with the increase of the average number of items per sequence. The running time of the algorithm was also tested on datasets with average length of maximal patterns ranging from 5 to 15. Each dataset had 100,000 customers with an average sequence length of 30, and the support threshold also set to 0.2.

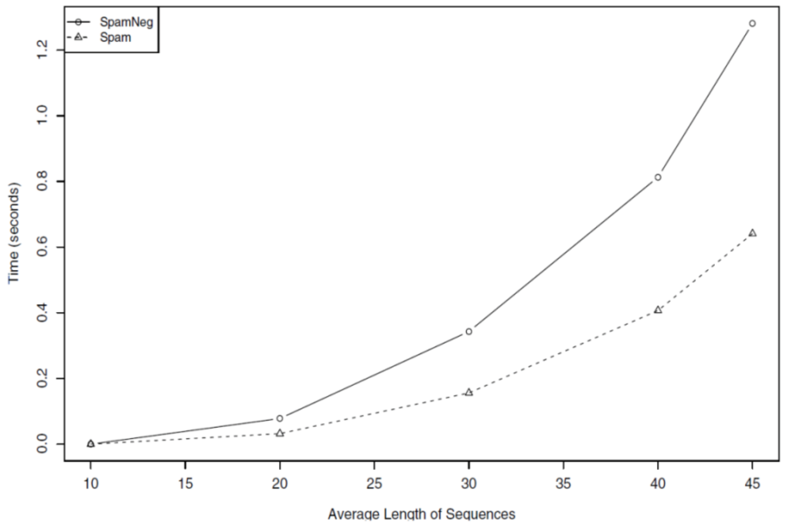


Figure 4-4 Scalability with the Number of Items per Sequence

Figure 4.5 shows that the running time decreases slightly with the increase of the average length of maximal patterns.

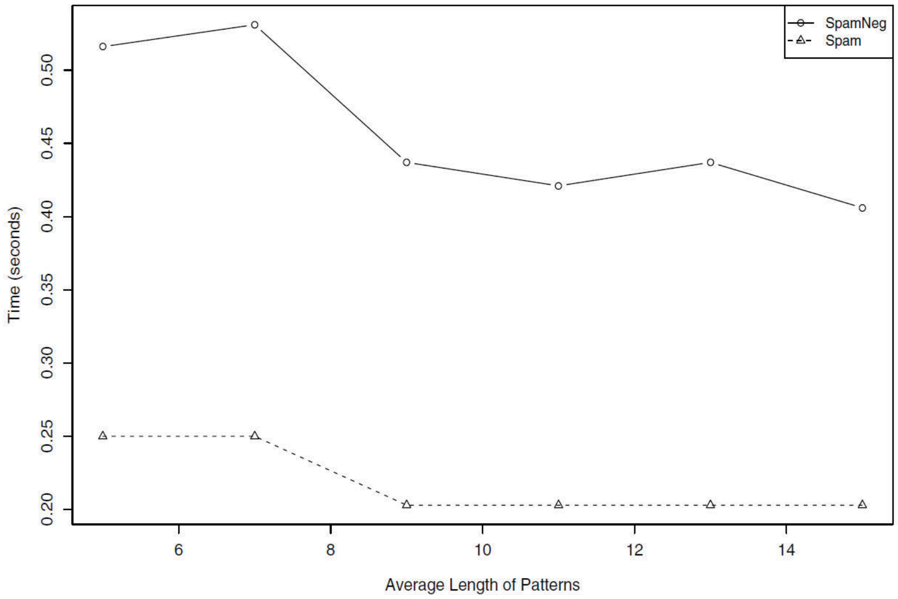


Figure 4-5 Scalability with the Average Length of Patterns

4.2.3 Conclusion

After defining and deriving the supports, confidences and lifts for negative sequential rules, an efficient algorithm is developed for mining event-oriented negative sequential rules based on the SPAM algorithm. Our algorithm has been tested on numerous synthetic datasets generated with IBM data generator, which shows its efficiency and scalability. The proposed algorithm can only find negative sequential rules with a single event on the right side. Sometimes it may be interesting to find more generalized negative sequential rules, which will be included in our future work. Moreover, negative sequential rules with time constraints will also be a part of our future research.

4.3 Mining Both Positive and Negative Impact-oriented Sequential Rules from Transactional Data

4.3.1 Mining Impact-Oriented Sequential Rules

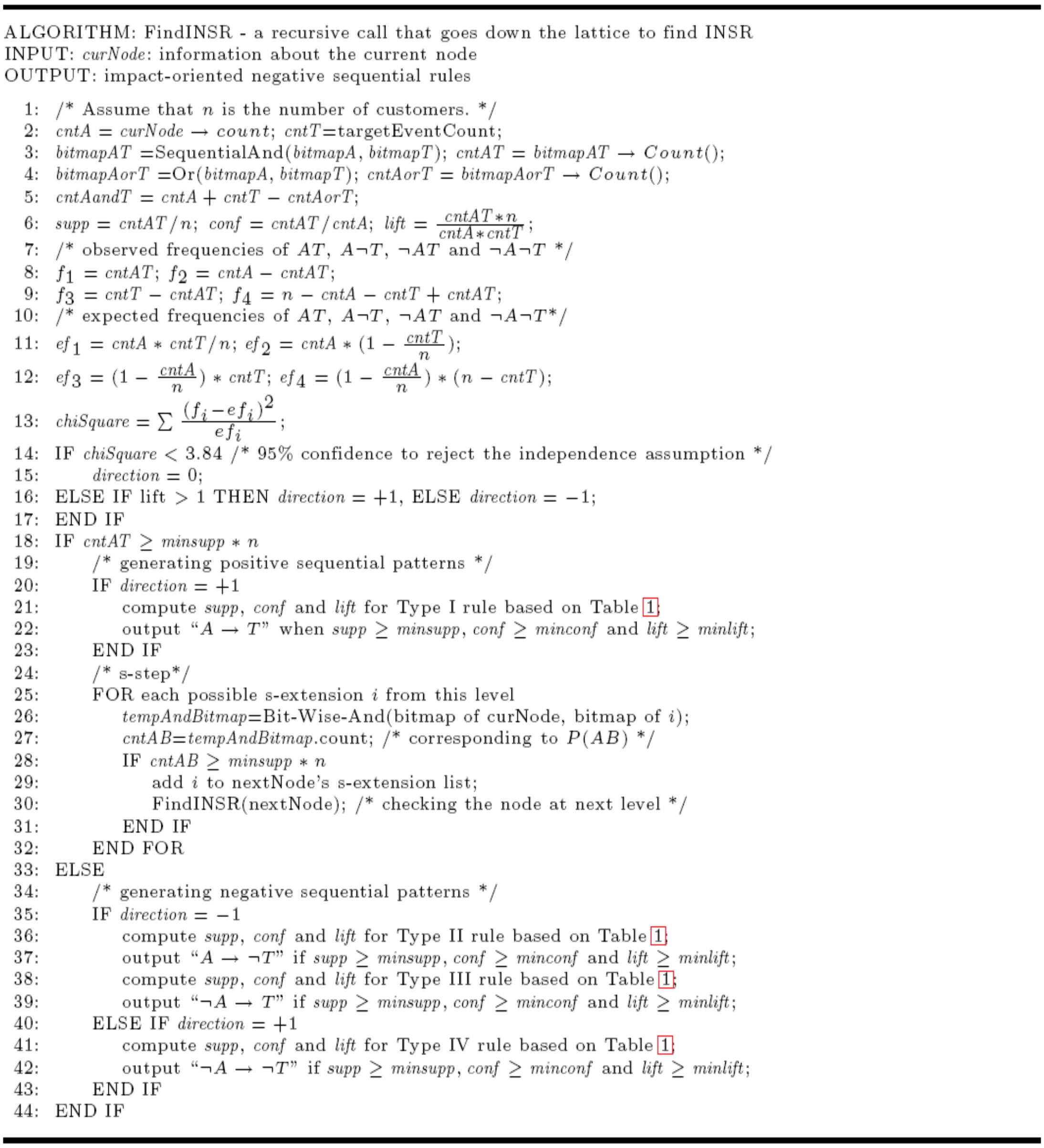
4.3.1.1 Algorithm for Mining Impact-Oriented Sequential Rules

To discover impact-oriented negative sequential rules, we once more use SPAM (Sequential Pattern Mining) [164] as a starting point, as it was demonstrated by Ayres et al. to be more efficient than SPADE and PrefixSpan [164]. SPAM is very efficient in that it uses bitmap to count the frequency of sequences. It searches the sequence lattice in a depth-first way, and candidates of longer sequences Sg are generated by append frequent items {i} to existing frequent sequences Sa. The candidate generation of SPAM is composed of two steps: S-step and I-step. The S-step appends i to S_a, which builds a longer sequence $S_g = S_a \bowtie i$. The I-step adds i to the last itemset of S_a, which builds a new sequence of the same length as S_a. In this work, we consider transaction with one item only and an element in the sequence is a single item, instead of an itemset. Therefore, only S-step from SPAM is used in our technique.

Figure 4.6 gives the pseudocode for finding impact-oriented negative sequential rules, which is based on the function "FindSequentialPatterns" from SPAM [164]. Lines 2-17 show the code for appending the target outcome to a sequential pattern and computing the chi-square and direction for the derived sequential rule. Lines 2-6 use bitmaps to compute the counts, support, confidence and lift for the sequential rule. Lines 7-17 compute the observed frequencies and expected frequencies, and then calculate chi-square and direction. Lines 19-23 generate positive sequential patterns. Lines 25-32 are the S-step of SPAM, which tries to extend the sequential pattern at current node by appending an additional item to it. Lines 34-43 generate three types of negative sequential patterns.

4.3.1.2 New Metrics for Impact-Oriented Sequential Rules

Two new metrics, contribution and impact, are designed as follows to select interesting impact-oriented sequential rules.



```
ALGORITHM: FindINSR - a recursive call that goes down the lattice to find INSR
INPUT: $curNode$: information about the current node
OUTPUT: impact-oriented negative sequential rules

/* Assume that $n$ is the number of customers. */
$cntA = curNode \rightarrow count$; $cntT$=targetEventCount;
$bitmapAT$ =SequentialAnd($bitmapA$, $bitmapT$); $cntAT = bitmapAT \rightarrow Count()$;
$bitmapAorT$ =Or($bitmapA$, $bitmapT$); $cntAorT = bitmapAorT \rightarrow Count()$;
$cntAandT = cntA + cntT - cntAorT$;
$supp = cntAT/n$; $conf = cntAT/cntA$; $lift = \frac{cntAT*n}{cntA*cntT}$;
/* observed frequencies of $AT$, $A\neg T$, $\neg AT$ and $\neg A\neg T$ */
$f_1 = cntAT$; $f_2 = cntA - cntAT$;
$f_3 = cntT - cntAT$; $f_4 = n - cntA - cntT + cntAT$;
/* expected frequencies of $AT$, $A\neg T$, $\neg AT$ and $\neg A\neg T$*/
$ef_1 = cntA * cntT/n$; $ef_2 = cntA * (1 - \frac{cntT}{n})$;
$ef_3 = (1 - \frac{cntA}{n}) * cntT$; $ef_4 = (1 - \frac{cntA}{n}) * (n - cntT)$;
$chiSquare = \sum \frac{(f_i - ef_i)^2}{ef_i}$;
IF $chiSquare < 3.84$ /* 95% confidence to reject the independence assumption */
    $direction = 0$;
ELSE IF lift $> 1$ THEN $direction = +1$, ELSE $direction = -1$;
END IF
IF $cntAT \geq minsupp * n$
    /* generating positive sequential patterns */
    IF $direction = +1$
        compute $supp$, $conf$ and $lift$ for Type I rule based on Table 1;
        output "$A \rightarrow T$" when $supp \geq minsupp$, $conf \geq minconf$ and $lift \geq minlift$;
    END IF
    /* s-step*/
    FOR each possible s-extension $i$ from this level
        $tempAndBitmap$=Bit-Wise-And(bitmap of curNode, bitmap of $i$);
        $cntAB$=$tempAndBitmap$.count; /* corresponding to $P(AB)$ */
        IF $cntAB \geq minsupp * n$
            add $i$ to nextNode's s-extension list;
            FindINSR(nextNode); /* checking the node at next level */
        END IF
    END FOR
ELSE
    /* generating negative sequential patterns */
    IF $direction = -1$
        compute $supp$, $conf$ and $lift$ for Type II rule based on Table 1;
        output "$A \rightarrow \neg T$" if $supp \geq minsupp$, $conf \geq minconf$ and $lift \geq minlift$;
        compute $supp$, $conf$ and $lift$ for Type III rule based on Table 1;
        output "$\neg A \rightarrow T$" if $supp \geq minsupp$, $conf \geq minconf$ and $lift \geq minlift$;
    ELSE IF $direction = +1$
        compute $supp$, $conf$ and $lift$ for Type IV rule based on Table 1;
        output "$\neg A \rightarrow \neg T$" if $supp \geq minsupp$, $conf \geq minconf$ and $lift \geq minlift$;
    END IF
END IF
```

Figure 4-6 Pseudocode for Discovering Impact-oriented Negative Sequential Rules

Definition 6 (Contribution). For a sequential rule P → T, where P is a sequential pattern, assume i to be the last item in P. The contribution of i to the occurrence of outcome T in rule P → T is

$$\text{contribution(i, P)} = \frac{\text{lift}(\text{P} \rightarrow \text{T})}{\text{lift}(\text{P} \setminus \text{i} \rightarrow \text{T})} \qquad (1)$$

where P \ i denotes the sequential pattern derived by removing i from P.

Definition 7 (Impact). *For the above rule and i, the impact of i on the outcome* in the rule is

$$\mathrm{impact}(\mathrm{i},\mathrm{P}) = \begin{cases} \mathrm{contribution}(\mathrm{i},\mathrm{P}) - 1 : & \text{if contribution} \geq 1 \\ \dfrac{1}{contribution(i,P)} - 1 : & otherwise \end{cases} \qquad (2)$$

Contribution shows how much the last item i in the rule contributes to the occurrence of the outcome *T*, and impact measures how much it can change the outcome. Both of them fall in [0,+∞).

4.3.2 Experimental Results

4.3.2.1 Performance and Scalability

Our designed algorithm (referred to as INSR) was implemented with C++ based on SPAM [164], and its performance and scalability was tested on synthetic datasets generated with IBM data generator [152]. All the tests were conducted on a PC with Intel Core 2 CPU of 1.86GHz, 2GB memory and Windows XP Pro. SP2. The number of items per transaction was set to one when generating data.

Our algorithm was first tested on a dataset with 50,000 customers, 40 items per sequence and the length of maximal patterns as 13. The minimum supports range from 0.2 to 0.7, and the results are shown in Figure 4.7a. From the figure, both INSR and Spam [164] run faster with larger minimum support, because the search space becomes smaller. Moreover, INSR runs faster than Spam, and the reason is that, when a pattern A is frequent and A __ T is infrequent, INSR doesn't search A's children nodes, but Spam continues checking all its descendants until it becomes infrequent.

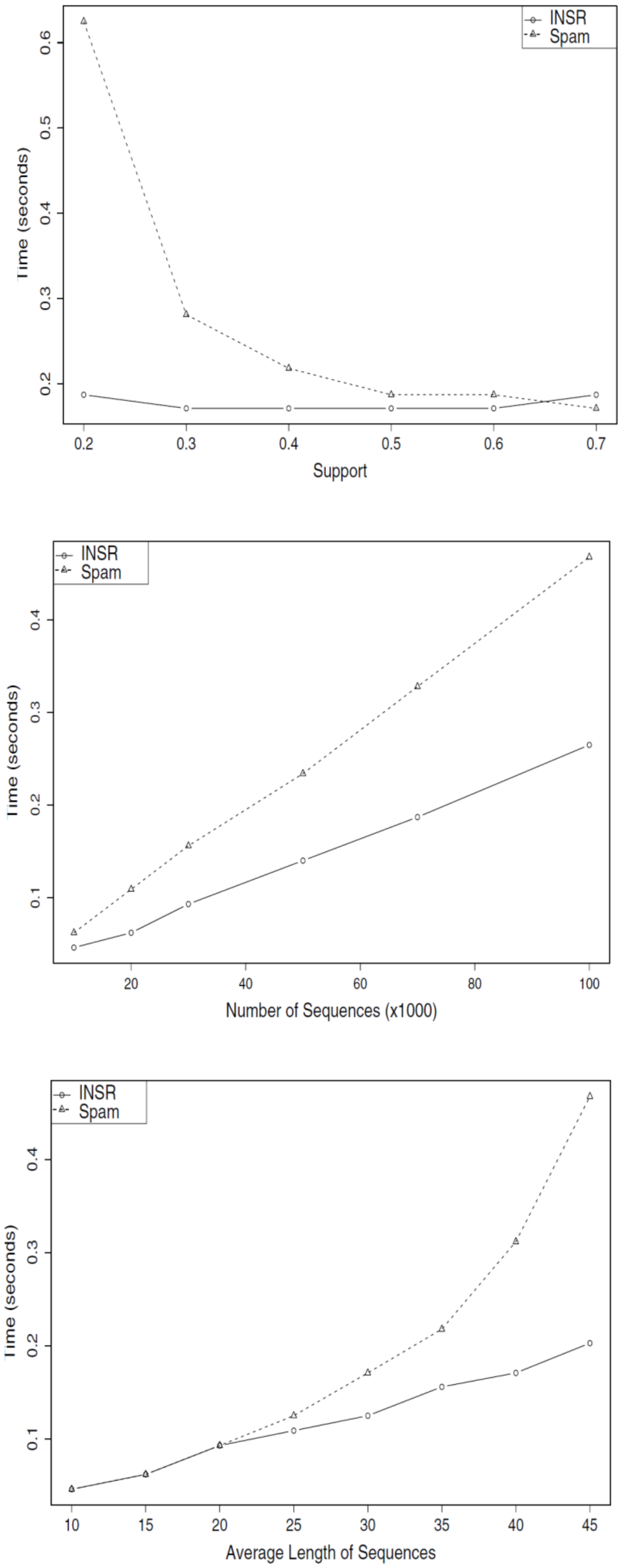


Figure 4-7 Scalability with (a) support; (b) the number of sequences; and (c) the length of sequences

The scalability with the number of sequences was tested on datasets with average sequence length as 30, length of maximal patterns as 11. The number of customers ranges from 10,000 to 100,000, and the support threshold is set to 0.3. Figure 4.7b shows the result of the above test. It's clear from the figure that INSR is linear with the number of sequences.

The running time with varying sequence lengths is shown Figure 4.7c, where the datasets used have 50,000 customers, with length of maximal patterns as 10, and the average sequence length ranging from 10 to 45. The support threshold is set to 0.3. The figure shows that the running time becomes longer with the increase of the average number of items per sequence and that INSR is almost linear with the length of sequences.

4.3.2.2 Selected Results in a Case Study

The proposed technique was applied to the real data from Centrelink, Australia. Centrelink is a Commonwealth Government agency distributing social welfare payments to entitled customers. For various reasons, customers on benefit payments or allowances sometimes get overpaid and these overpayments lead to debts owed to Centrelink. We used impact-oriented negative sequential rules to find the relationship between transactional activity sequences and debt occurrences, and also find the impact of additional activities on debt occurrence.

A sample of historical transactional data from July 2007 to February 2008 were used for the analysis. After data preprocessing, 15,931 sequences were constructed.

Minimum support was set to 0.05, that is, 797 out of 15,931 sequences. There are 2,173,691 patterns generated and the longest pattern has 16 activities. Some selected sequential rules are given in Table 4.3, where "DEB" stands for debt and the other codes are activities. "Direction" shows whether the pattern is positively (+1) or negatively (-1) associated with debt occurrence.

Table 4-3 Selected positive and negative sequential rules

Type	Rule	Supp	Conf	Lift	Direction
I	REA ADV ADV→DEB	0.103	0.53	2.02	+1
I	RPR ANO→DEB	0.111	0.33	1.25	+1
I	STM PYI→DEB	0.106	0.30	1.16	+1
II	MND→ ¬DEB	0.116	0.85	1.15	-1
II	REA PYR RPR RPT→ ¬DEB	0.176	0.84	1.14	-1
II	REA CRT DLY→ ¬DEB	0.091	0.83	1.12	-1
III	¬{PYR RPR REA STM}→DEB	0.169	0.33	1.26	-1
III	¬{PYR CCO}→DEB	0.165	0.32	1.24	-1
III	¬{PLN RPT}→DEB	0.212	0.28	1.08	-1
IV	¬{REA EAN}→ ¬DEB	0.650	0.79	1.07	+1
IV	¬{DOC FRV}→ ¬DEB	0.677	0.78	1.06	+1

Figure 4.8 on page 66 shows an example of discovered growing sequential pattern,

$$\begin{cases} ADV \rightarrow DEB \\ ADV, ADV \rightarrow DEB \\ ADV, ADV, CCO \rightarrow DEB \end{cases} . \tag{3}$$

Each point in every chart gives the value for the sequential pattern from the first activity to the corresponding activity. All four charts in Figure 4.8 show the growth from “ADV” to “ADV ADV” and “ADV ADV CCO”. ADV increases the probability of debt occurrence, because its confidence in debt occurrence is 0.395, 1.5 times the likelihood of debt occurrence in the whole population (see the first chart). There are 18% of all sequences supporting that ADV is followed by debt (see the second chart). As shown in the third chart, the two ADVs contributes to debt occurrence, but CCO contributes negatively, as its contribution is less than one. The impacts of two ADVs on outcome are different, with the first one having larger impact (see the fourth chart).

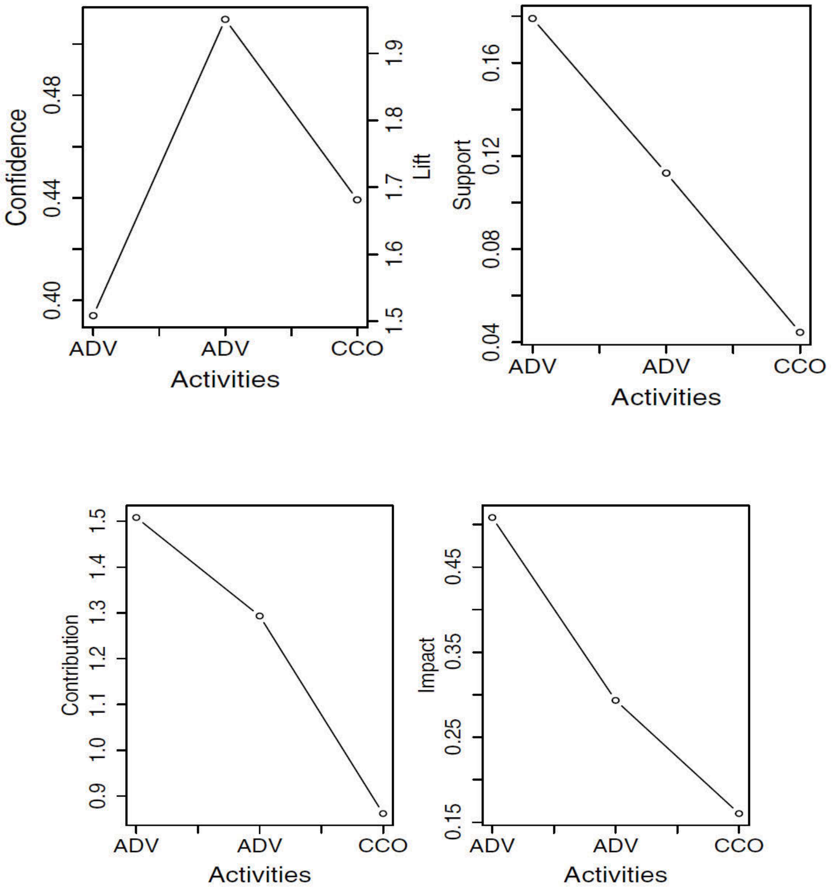


Figure 4-8 A Growing Sequential Pattern “ADV ADV CCO”

4.3.3 Conclusions

We have defined impact-oriented negative sequential rules and have designed an efficient algorithm for mining such sequential rules. We have also designed two metrics, contribution and impact, to measure the effect of an item on the outcome, which help to select interesting growing sequential patterns. A case study has been presented to show the effectiveness of the proposed technique.

5 Predict Debt-related Social Security Activity Sequences

5.1 Problem Statement of Sequence Classification

Let S be a sequence database, in which each sequence is an ordered list of elements. These elements can be either simple items from a fixed set of items, or itemsets, that is, non-empty sets of items. The list of elements of a data sequence s is denoted by $< s_1, s_2, \ldots\ldots , s_n >$, where s_i is the ith element of s.

Consider two sequences $s =< s_1, s_2, \ldots\ldots , s_n >$ and $t =< t_1, t_2, \ldots\ldots , t_m >$. We say that s is a subsequence of t if s is a “projection” of t, derived by deleting elements and/or items from t. More formally, s is a subsequence of t if there exist integers $j_1 < j_2 < . . . < j_n$ such that $s_1 \subseteq t_{j1}$, $s_2 \subseteq t_{j2}, . . ., s_n \subseteq t_{jn}$. Note that for sequences of simple items the above condition translates to $s_1 = t_{j1}, s_2 = t_{j2}, . . ., s_n = t_{jn}$. A sequence t is said to contain another sequence s if s is a subsequence of t, in the form of $s \subseteq t$.

5.1.1.1 Frequent Sequential Patterns

The number of sequences in a sequence database *S* containing sequence *s* is called the *support* of *s*, denoted as *sup(s)*. Given a positive integer *min_sup* as the support threshold, a sequence *s* is a frequent sequential pattern in sequence database *S* if $sup(s) \geq min\_sup$. The sequential pattern mining is to find the complete set of sequential patterns with respect to a given sequence database S and a support threshold min_sup.

5.1.1.2 Classifiable Sequential Patterns

Let T be a finite set of class labels. A sequential classifier is a function

$$F : S \rightarrow T \qquad (1)$$

In sequence classification, the classifier *F* is built on the base of frequent classifiable sequential patterns *P*.

Classifiable Sequential Patterns (CSP) are frequent sequential patterns for the sequential classifier in the form of $p_a \Rightarrow \tau$, where p_a is a frequent pattern in the sequence database S.

Based on the mined classifiable sequential patterns, a sequential classifier can be formulized as

$$\mathcal{F} : s \xrightarrow{\mathcal{P}} \tau. \qquad (2)$$

That is, for each sequence $s \in S$, F predicts the target class label of s based on the sequential classifier built with the classifiable sequential pattern set P. Suppose we have a classifiable sequential pattern set P. A sequence instance s is said to be covered by a classifiable sequential pattern $p \in P$ if s contains the antecedent p_a of the classifiable sequential pattern p.

5.2 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.

5.2.1 Discriminative Sequential Patterns

Given a sequence dataset S and a set of target classes 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) [166].

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 5.1), CCR is defined as

$$CCR(p_a \to \tau) = \frac{c\widehat{or}r(p_a \to \tau)}{c\widehat{or}r(p_a \to \neg\tau)} = \frac{a \cdot (c+d)}{c \cdot (a+b)}, \qquad (3)$$

where corr(p_a→ τ) is the correlation between p_a and the target class τ, defined as

$$c\widehat{or}r(p_a \to \tau) = \frac{sup(p_a \cup \tau)}{sup(p_a) \cdot sup(\tau)} = \frac{a \cdot n}{(a+c)\cdot(a+b)}. \qquad (4)$$

CCR falls in [0,+∞). 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.

Table 5-1 Feature-Class Contingency Table

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

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

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

where M is the maximum W_s of all rules where CCR ≠ 0.

5.2.2 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 [111,116].

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 P.

3) Pruning patterns in the obtained classifiable sequential pattern set with the pattern pruning algorithm in [167]. 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 [168,167]. 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.

5.2.3 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.

5.2.3.1 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 5.2. 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”.

Table 5-2 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

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 5-3 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

5.2.3.2 Results of Negative Sequential Pattern Mining

Our previous technique on negative sequential rules [116] 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 5.3, 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.

5.2.3.3 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 5.3, 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 →¬DEB can be taken as a positive rule A → 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 5.5–5.8, 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 obtained two sets of sequential patterns, "PS05" and "PS10". The numbers of the four types of patterns are shown in Table 5.4. 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 Ws. 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.

Table 5-4 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

Following this, 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 5.5-5.8. In the four tables, the first rows show the number of patterns used in the classifiers. In Tables 5.7 and 5.8, 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 5.5, 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 5.6, 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 5.7 and 5.8, 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 5.9). 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 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)

Table 5-5 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

Table 5-6 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 5-7 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 5-8 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 5-9 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

5.3 Debt Detection in Social Security by Adaptive Sequence Classification

5.3.1 Discriminative Frequent Patterns Boosting

Given a dataset, the more samples a pattern can correctly classify, the more discriminative the pattern is on the dataset. In other words, the more samples a pattern incorrectly classifies, the less discriminative the pattern is on the dataset. To make it more statistically significant, the definitions of positive contribution ability and negative contribution ability are given as follows.

Definition 5.2 (Positive Contribution Ability). Given a dataset *S*, the Positive Contribution Ability *(PCA)* of pattern *P* is the proportion of samples that can be correctly classified by P out of all the samples in dataset *S*.

Definition 5.3 (Negative Contribution Ability). Given a dataset *S*, the Negative Contribution Ability *(NCA)* of pattern *P* is the proportion of samples that are incorrectly classified by P out of all the samples in the dataset S.

For a classifiable sequential pattern *P* in the form of $p_a \Rightarrow \tau$, *PCA* of *P* on *S* can be denoted as

$$PCA_S(P) = \frac{\|\{s|p_a \subseteq s \wedge s \in S_\tau\}\|}{\|S\|}, \qquad (3)$$

and *NCA* of pattern *P* on *S* can be denoted as

$$NCA_S(P) = \frac{\|\{s|p_a \subseteq s \wedge s \in S_{\neg\tau}\}\|}{\|S\|}, \qquad (4)$$

where S_τ and $S_{\neg\tau}$ represent the subsets of S in which samples are of class τ and are not of class τ, respectively.

Above all, PCA and NCA describe the classification ability of patterns on a given dataset. In order to enhance classification performance, it is intuitive to boost the patterns with higher PCA and lower NCA, while depress those with lower PCA and higher NCA. Thereafter, a measure of Contribution Weight is proposed to measure the discriminative power that a pattern contributes to the classification on a dataset.

Definition 5.4 (Contribution Weight). Given a dataset *S*, Contribution Weight of a classifiable sequential pattern *P* is the ratio of Positive Contribution Ability $PCA_S(P)$ on *S* and Negative Contribution Ability $NCA_S(P)$ on *S*. It can be denoted as

$$CW_S(P) = \frac{PCA_S(P)}{NCA_S(P)} = \frac{\|\{s|p \subseteq s \wedge s \in S_\tau\}\|}{\|\{s|p \subseteq s \wedge s \in S_{\neg\tau}\}\|}.$$

The proposed measure of contribution weight tells the relative discriminative power of a classifiable sequential pattern on a given dataset, which is based on the classification performance of the pattern on the dataset. According to the definition, contribution weight has following characters.

– The greater the value of contribution weight is, the more discriminative a pattern is on a given dataset, and vice versa.

– Contribution weight is a measure with regard to a dataset on which classification performance is evaluated.

– Contribution weight is independent of the algorithm that is used for classifiable sequential pattern mining, and it does not matter which interestingness measure is used for classification.

Therefore, we introduce contribution weight as a factor to boost the discriminative frequent patterns on a certain dataset. The term of Boosted Interestingness Measure is defined as follows.

Definition 5.5 (Boosted Interestingness Measure). For a classifiable sequential pattern *P* with an interestingness measure *R*, the corresponding Boosted Interestingness Measure on dataset *S* is denoted as

$$R_S^* = R \times CW_S(P). \qquad (6)$$

In other words, boosted interestingness measure of a pattern can be regarded as a weighted interestingness measure, and the weight tells how much contribution the corresponding pattern can make to the classification on the given dataset. Patterns that are more discriminative on a given dataset are strongly boosted by higher contribution weights, and vice versa. From this point of view, boosted interestingness measure adjusts the original interestingness measure so as to make it indicating the discriminative ability of classifiable sequential patterns on the given dataset more vividly.

5.3.2 Adaptive Sequence Classification Framework

In order to catch up with the pattern variation over time, an adaptive sequence classification framework is introduced in this section. The main idea of the adaptive framework is to include the latest pattern into the classifier with the proposed boosted interestingness measure, so as to improve the classification performance on dataset of near future.

As illustrated in Figure 5.1, the initial classifiable sequential pattern set CSP_0 is extracted from the dataset DS_0, and then is used to perform prediction/classification on coming dataset DS_1 and get the predicted labels L'$_1$. Once L_1, the real class labels of dataset DS_1, is available, interestingness measure of the classifier CSP0 could be refined and CSP_0 evolves into CSP_1 with boosted interestingness measure, which brings the timely trends of patterns in dataset DS_1 into the classification model. The boosted classifier will be applied to continuously coming dataset for prediction/classification. The procedure goes on as dataset updates all along, which is generalized in Algorithm 1. The boosted classifier CSP_i, i = 1, 2, ... not only takes the latest pattern variation into the classification model, but also tracks the evolvement of the patterns ever since the initial classifier is built. Therefore, the performance of classification is expected to outperform that of the initial classifier.

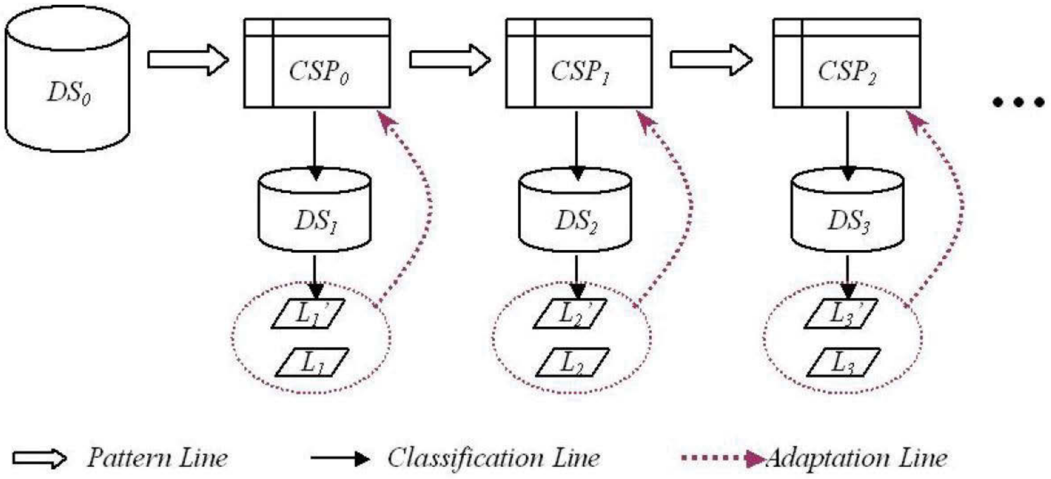
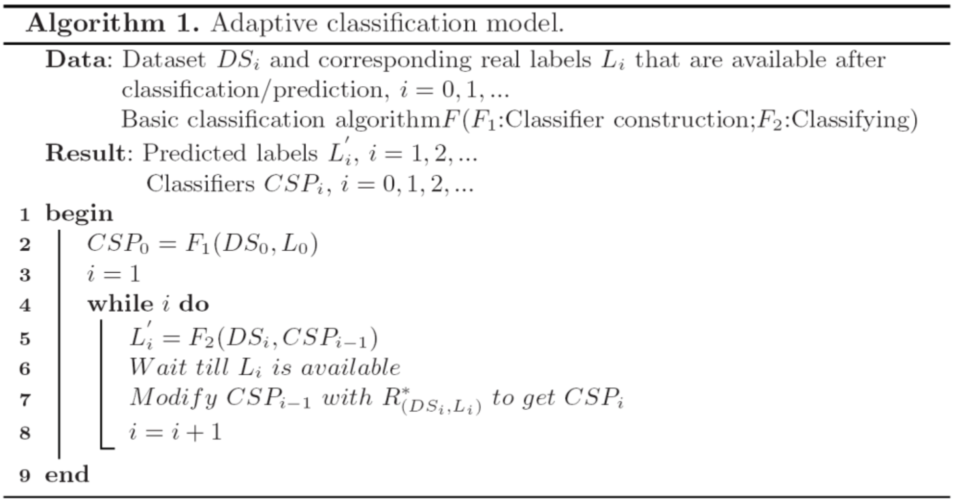


Figure 5-1 Architecture of Adaptive Sequence Classification



```
Algorithm 1. Adaptive classification model.
Data: Dataset $DS_i$ and corresponding real labels $L_i$ that are available after
      classification/prediction, $i = 0, 1, ...$
      Basic classification algorithm$F$($F_1$:Classifier construction;$F_2$:Classifying)
Result: Predicted labels $L_i^{'}$, $i = 1, 2, ...$
        Classifiers $CSP_i$, $i = 0, 1, 2, ...$
begin
    $CSP_0 = F_1(DS_0, L_0)$
    $i = 1$
    while $i$ do
        $L_i^{'} = F_2(DS_i, CSP_{i-1})$
        $Wait\ till\ L_i\ is\ available$
        $Modify\ CSP_{i-1}\ with\ R^*_{(DS_i,L_i)}\ to\ get\ CSP_i$
        $i = i + 1$
end
```

Figure 5-2 Adaptive Classification Model

Since the adaptive model is based on boosted interestingness measure, it inherits the properties of boosted interestingness measure congenitally. To be more precise, it is independent of interestingness measure and classifiable sequence mining method.

5.3.3 Case Study

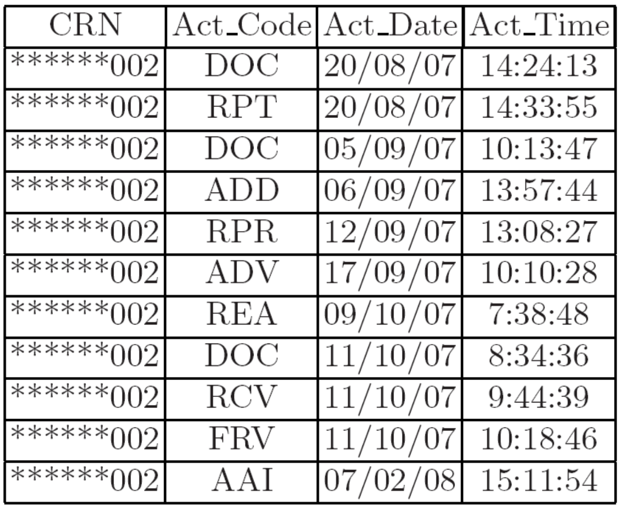
The proposed algorithm has been applied in a real world business application in Centrelink, Australia. The purpose of the case study is to predict and further prevent debt occurrence based on customer transactional activity data. In this section, the dataset used for debt prediction in Centrelink is described firstly. Then a pre-experiment is given to evaluate the effectiveness of discriminative pattern boosting strategy, followed by the experimental results of adaptive sequence classification framework.

5.3.3.1 Data Description

The dataset used for sequence classification is composed of customer activity data and debt data. In Centrelink, every single contact (e.g., a circumstance change) of a customer may trigger a sequence of activities running. As a result, large volumes of activity based transactions are encoded into 3-character “Activity Code” and recorded in activity transactional files. In the original activity transactional table, each activity has 35 attributes, in which 4 attributes are used in the case study.

These attributes are “CRN” (Customer Reference Number) of a customer, “Activity Code”, “Activity Date” and “Activity Time”, as shown in Table 5.10. We sort the activity data according to “Activity Date” and “Activity Time” to construct the activity sequence. The debt data consist of the “CRN” of the debtor and “Debt Transaction Date”. In our case study, only the activities of a customer before the occurrence of his/her debt are kept for the sequence classification.

Table 5-10 Centrelink Data Sample



CRN	Act_Code	Act_Date	Act_Time
******002	DOC	20/08/07	14:24:13
******002	RPT	20/08/07	14:33:55
******002	DOC	05/09/07	10:13:47
******002	ADD	06/09/07	13:57:44
******002	RPR	12/09/07	13:08:27
******002	ADV	17/09/07	10:10:28
******002	REA	09/10/07	7:38:48
******002	DOC	11/10/07	8:34:36
******002	RCV	11/10/07	9:44:39
******002	FRV	11/10/07	10:18:46
******002	AAI	07/02/08	15:11:54

5.3.3.2 Effectiveness of Boosting Discriminative Patterns

In order to evaluate the effectiveness of discriminative patterns boosting, two groups of experiments are presented in this section. In both groups, we compare the performance of classification which uses discriminative pattern boosting strategy with that does not boost discriminative patterns. In group (a), the activity sequence data generated from Jul. 2007 to Oct. 2007 are used. After data cleaning, there are 6, 920 activity sequences including 210, 457 activity records used. The dataset is randomly divided into the following 3 portions.

– Training data(60%): To generate the initial classifier.

– Evaluation data(20%): To refine classifier.

– Test data(20%): To test the performance of classification.

While in group (b), some data generated in Nov. 2007 is added to the evaluation data and test data, expecting to include some pattern variation.

According to the property of contribution weight, the boosted interestingness measure is independent of basic classification. Therefore, we use the classification algorithm proposed in our previous work [108] to generate the initial classifier on the training dataset. And we use confidence as the base interestingness measure. For classification which uses boosting strategy, the evaluation dataset is used to refine the initial classifier, and the refined classifier is evaluated on the test dataset. While for the classification that does not boost discriminative patterns, we combine training data and evaluation data to generate the initial classifier, and then apply the initial classifier to the test dataset for debt prediction.

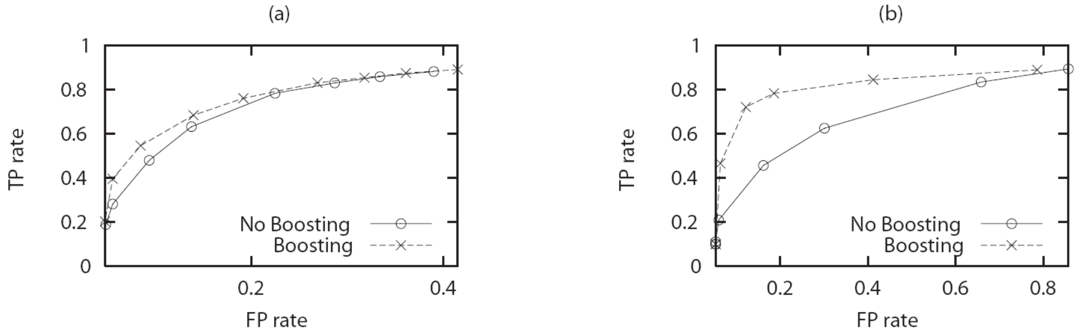


Figure 5-3 Effectiveness of Discriminative Patterns Boosting

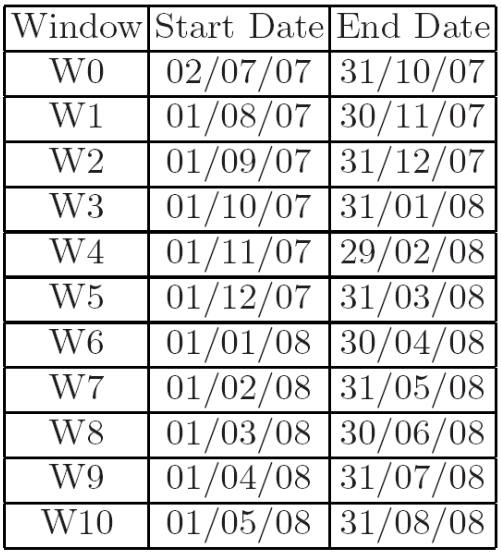
ROC curve (Receiver Operating Characteristic) is used to plot the fraction of true positives vs. the fraction of false positives of each classifier. The best possible prediction method would yield a point in the upper left corner or coordinate (0,1) of the ROC space, representing 100% sensitivity (no false negatives) and 100% specificity(no false positives). Therefore, the more close to the upper left corner the curve is, the better the classification method is. As illustrated in Fig. 5.2, the boosted classifier outperforms the classifier without boosting in both experiments. In group (a), training data, evaluation data and test data all come from the dataset generated in the same time period. By boosting discriminative patterns with evaluation data, classification power of initial classifier is refined by boosting discriminative patterns and depressing less discriminative patterns, so it outperforms the classification without boosting. As for group (b), since some new data generated in different time period is added to the evaluation data and test data, some pattern variation might be included in the corresponding dataset. In this circumstance, the proposed boosting strategy notices the pattern variation in the updated dataset, refines the interestingness measure of the classifiers with evaluation data, and performs much better in the test data than the classifier without boosting.

In all, the discriminative pattern boosting strategy improves the classification performance, especially when the sequence data evolves with pattern variation.

5.3.3.3 Performance of Adaptive Sequence Classification Framework

In this subsection, we will evaluate the adaptive sequence classification framework on the sequence datasets obtained with a sliding window applied on the activity sequence data. After applying sliding window on the sequences generated from Jul. 2007 to Aug. 2008, we get 11 windows listed in Table 5.11.

Table 5-11 Data Windows



Window	Start Date	End Date
W0	02/07/07	31/10/07
W1	01/08/07	30/11/07
W2	01/09/07	31/12/07
W3	01/10/07	31/01/08
W4	01/11/07	29/02/08
W5	01/12/07	31/03/08
W6	01/01/08	30/04/08
W7	01/02/08	31/05/08
W8	01/03/08	30/06/08
W9	01/04/08	31/07/08
W10	01/05/08	31/08/08

Following the framework proposed in Section 4, the classification in our previous work [108] is firstly applied on W_0 and the initial classifier CSP_0 is generated. By discriminative pattern boosting with W_1, CSP_0 is refined to CSP_1 and then is applied to make debt prediction on W_2. Here we still use confidence as the base interestingness measure. The debt prediction performance on W_2 is illustrated in the first graph in Fig. 5.3. Thereafter, CSP_1 is boosted with sequence data in W_2, and the generated CSP_2 is applied on W3 to predict debt occurrence. As the procedure goes on continuously, the debt prediction performance on all the following windows are listed in Fig. 5.3, which is represented by the ROC curves labelled

Adaptation all along. In order to evaluate the performance of adaptive sequence classification framework, debt prediction on each window is also performed with initial classifier CSP_0, whose performance is denoted by the ROC curves labelled No adaptation. According to Fig. 5.3, we can tell that the proposed adaptive framework outperforms the initial classifier in debt prediction on continuously coming datasets. Since the classifier is continuously updated with the latest data, it catches up the pattern variation in the new dataset and then works well on the debt prediction on the oncoming dataset. Meanwhile, we apply CSP_1, which is boosted once based on initial classifier, to each of the windows and get the performance denoted by the curves labelled Adaptation once. The classifier boosted once still outperforms the initial classifier. While it does not contain the pattern information in the latest datasets, its performance is always worse than that of Adaptation all along strategy.

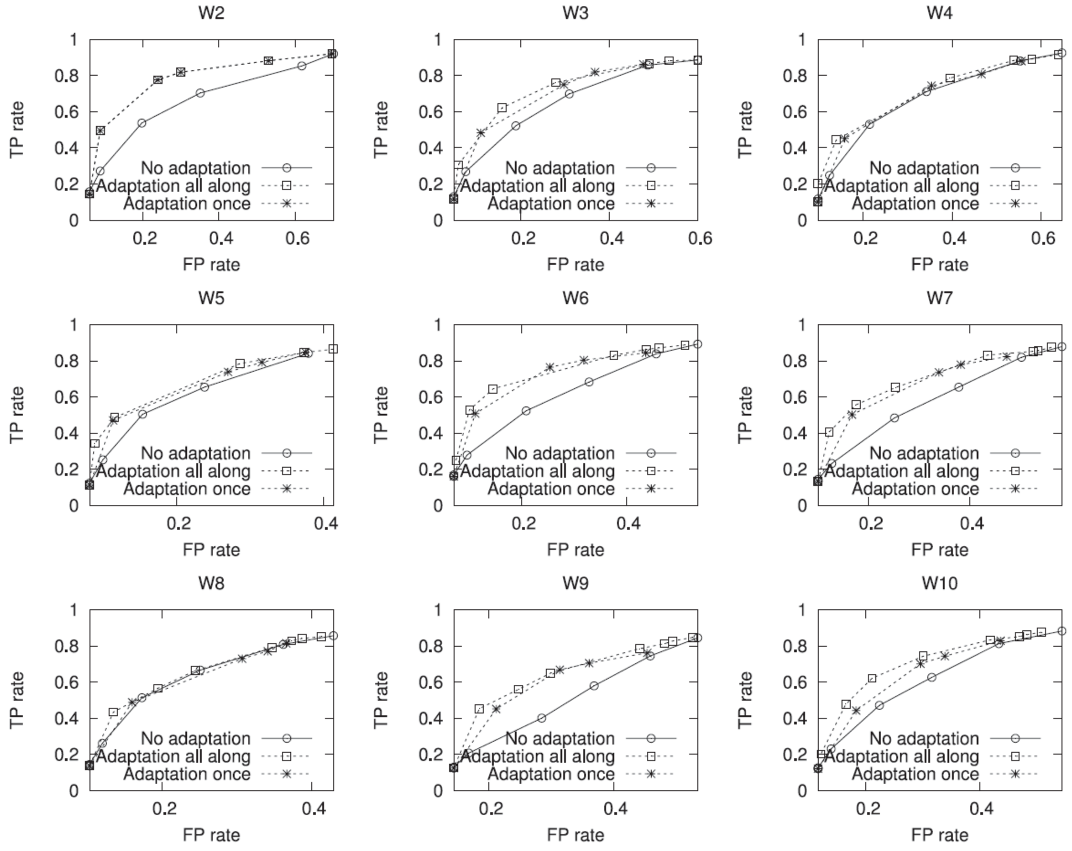


Figure 5-4 RoC curves of Adaptive Sequence Classification Framework

Above all, the conclusion could be drawn that our proposed adaptive sequence classification framework updates the classifier with new data, includes the sequence pattern variation in the new data, and performs effectively on the continuously arriving data.

5.4 Customer Activity Sequence Classification for Debt Prevention in Social Security

5.4.1 Interestingness Measure

Suppose there is a pattern A ⇒ B, there are three key properties (KP) and five other properties (OP) for a good interestingness measure (M) [169]. Basically, a good measure should satisfy the following three key properties:

- KP1: M = 0 if A and B are statistically independent;
- KP2: M monotonically increases with P(A,B) when P(A) and P(B) remain the same; and
- KP3: M monotonically decreases with P(A) (or P(B)) when the rest of the parameters (P(A,B) and P(B) or P(A)) remain unchanged.

Furthermore, there are five other properties. A measure should satisfy them or not depending on different conditions. The five properties are:

· OP1: symmetry under variable permutation;

· OP2: row/column scaling invariance;

· OP3: antisymmetry under row/column permutation;

· OP4: inversion invariance; and

· OP5: null invariance.

In this thesis, we aim to find the sequential patterns that either positively or negatively relate to a class. To this end, the interestingness measures should satisfy three key properties plus OP_3, that is, they should distinguish positive and negative correlations of a table.

The Class Correlation Ratio (CCR)[166] is used as the principal interestingness measure since it meets the above requirements. The Class Correlation Ratio (CCR) can be defined given a contingency table shown in Table 5.12.

Table 5-12 2 by 2 Feature-Class Contingency Table

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

Class correlation ratio is to measure how correlated the sequential pattern p_a is with the target class τ compared to negative class ¬τ. It employs the following formula –

$$\begin{aligned} CCR(p_a \rightarrow \tau) &= \frac{\hat{corr}(p_a \rightarrow \tau)}{\hat{corr}(p_a \rightarrow \neg\tau)} && (3) \\ &= \frac{a \cdot (c+d)}{c \cdot (a+b)}. && (4) \end{aligned}$$

and here we display the correlation between pa and the target class τ –

$$\begin{aligned} \hat{corr}(p_a \rightarrow \tau) &= \frac{sup(p_a \cup \tau)}{sup(p_a) \cdot sup(\tau)} && (5) \\ &= \frac{a \cdot n}{(a+c) \cdot (a+b)}, && (6) \end{aligned}$$

CCR falls in [0,+∞). CCR = 1 means the antecedent is independent of the target class. CCR < 1 means the antecedent is negatively correlated with the target class. CCR > 1 means the antecedent is positively correlated with the target class. Obviously, CCR can distinguish whether a pattern is positively and negatively correlated to a target class. Also, it is a asymmetric measure to differentiate the target class from antecedent sequential patterns.

5.4.2 Sequence Classification

Given a sequence database S and a set of target class T, the conventional method to build sequential classifier is to mine for the complete set of patterns with respect to a given support threshold min sup. Then, a number of processes are adopted to work on the large sequential pattern set to select the discriminative patterns for the classification.

In our algorithm we do not follow conventional algorithms to mine for the complete set of classifiable sequential pattern set. Instead, we build the sequential classifier in a hierarchical way as shown in Fig. 5.5.

The outline of the hierarchical sequence classification algorithm is as follows:

1) Applying sequential pattern mining algorithm on the input dataset. Instead of calculating support and confidence of each candidate pattern, the algorithm in this paper calculates the frequency of each classifiable sequential pattern and the corresponding CCR. Since CCR = 1 means the antecedent is independent of the target class, only the patterns with CCR > 1 + m_1 or CCR < 1−m_2 are selected as the candidate classifiable sequential patterns. Here m1 and m_2 are predefined margins. Aggressive strategy is used in our frequent sequential pattern mining stage, which is illustrated in Subsection 5.1.

2) After the sequential patterns are discovered, pattern pruning is implemented on the classifiable sequential pattern set. We follow the pattern pruning algorithm in [167]. The only difference is, in our algorithm, CCR is used as the measure for pruning instead of confidence. The brief introduction of our pruning algorithm is shown in Subsection 5.2.

3) Conduct serial coverage test following the ideas in [167,168]. The patterns which can correctly cover at least one training sample in the serial coverage test form the first level of sequential classifier. Please see Subsection 5.3 for the description of the serial coverage test.

4) Since aggressive pattern mining strategy is used in sequential pattern mining step, only a small number of classifiable sequential patterns are discovered at the first level. Hence in the serial coverage test, a large portion of training samples may not be covered by the mined classifiable sequential patterns. These training samples are fed back to Step 1). With updated parameters, sequential pattern mining is again implemented. After pattern pruning and coverage test (Step 2) to Step 3)), the samples that still cannot be covered are fed back for sequential pattern mining until the predefined thresholds are reached or all samples are covered. The classifiable sequential patterns mined from each loop form the sequential classifier at each level.

5) The final sequential classifier is the hierarchical one consisting of the above sub-classifiers at different levels.

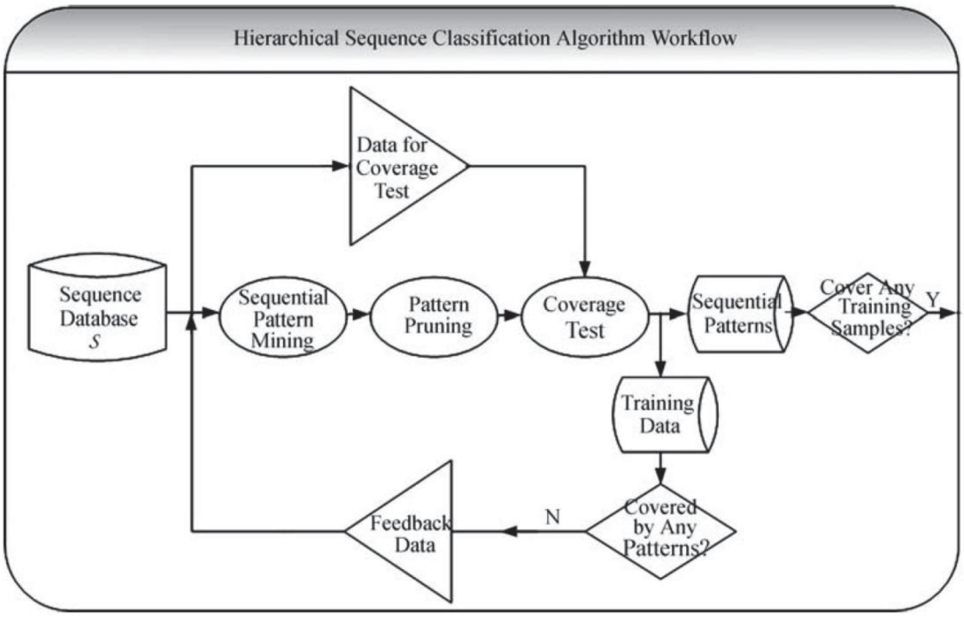


Figure 5-5 Hierarchical sequence classification algorithm

5.4.2.1 Sequential Pattern Mining

It is known that support is an anti-monotonic measure. The monotonics can dramatically reduce the searching space in frequent pattern mining algorithms. However, the general idea of a pattern being correlated to a target class is not anti-monotonic. To avoid examining the entire space, we use search strategies that ensure the concept of being potentially interesting is antimonotonic. That is, $p_a \rightarrow c$ might be considered as potentially interesting if and only if all $\{p'_a \rightarrow c | p'_a \subset p_a\}$ have been found to be potentially interesting. In this algorithm, we select a new item in such a way that it makes a significant positive contribution to the pattern, when compared to its all generalisations. The pattern $p_a \rightarrow c$ is potentially interesting only if the test passes for all generalisations. This effectively tells us that, when compared to the generalisations, all of the items in the antecedent of the pattern make a significant positive contribution to the patterns associated with the target class. This technique prunes the search space most aggressively, as it performs $|p_a|$ tests per rule, where $| \cdot |$ is the length of a pattern.

5.4.2.2 Pattern Pruning

In this thesis, two pattern pruning techniques are used to reduce the size of the mined sequential pattern set. The first technique is redundancy removal. We use general and high ranking patterns to prune more specific and low ranking patterns. Here the ranking is based on the weighted score which is defined as follows:

$$W_s = \begin{cases} CCR, & CCR > 1, \\ \frac{1}{CCR}, & CCR < 1, \\ M, & CCR = 0, \end{cases} \qquad (7)$$

where M is a predefined integer for the maximum value of the CCR in the algorithm.

Suppose two sequences p_i and p_j are in the mined sequential pattern set P. If the following conditions are met, the pattern p_j is pruned.

$$\begin{cases} p_i \subseteq p_j, \\ W_s^{p_i} > W_s^{p_j}. \end{cases}$$

The second pruning technique is significance testing. For each pattern $p_a \rightarrow c$, we test whether p_a is significantly correlated with c by χ^2 testing. Only the χ^2 value of a pattern greater than a threshold (in this thesis, 3.84) is kept for further processing. All the other patterns are pruned.

5.4.2.3 Coverage Test

The serial coverage test is invoked after patterns have been ranked as above. Only the sequential patterns that cover at least one training sample not covered by a higher ranked pattern are kept for later classification. For each sorted sequential pattern starting from the top ranked one s_1, a pass over the training data set to find all objects that match s_1 is performed. Once training objects covered by s_1 are located, they are removed from the training data set and s_1 is inserted into the sub-classifier. The process repeats for the remaining ordered patterns until all training objects are covered or all sorted patterns have been checked. If a pattern is unable to cover at least a single training object, then it will be discarded.

5.4.2.4 Weighted Classifier

After serial coverage test, we have a set of sequential patterns to build the sub-classifier at each level of the final classifier. In this paper, we follow two strategies to build each sub-classifier as follows.

- Highest weighted score ($CCR_{highest}$). Given a sequence instance s, the class label corresponding to the classifiable sequential pattern with highest weighted score is assigned to s.
- Multiple weighted scores (CCR_{multi}). Given one sequence instance s, all the classifiable sequential patterns at one level covering s are extracted. It is not difficult to compute the sum of the weighted score corresponding to each target class. The class label corresponding to the largest weighted score sum is assigned to s.

5.4.2.5 Classifier Testing

When the hierarchical classifier is built, it can be used for the prediction of the target class on a sequence database, which is also conducted in a hierarchical way. Suppose there is a sequence instance s. Firstly, sub-classifier on the first level is used to test on s. If s can be covered by the sub-classifier, the target class of s is predicted following the proposed algorithms $CCR_{highest}$ or CCR_{multi}. Otherwise s is input to the next level to test whether it is covered by the next sub-classifier until the last level. Since the prediction is also implemented in a hierarchical way, the coverage test is not needed to be done on the whole classifiable sequential pattern set. Hence the efficiency of prediction is also improved in our algorithm.

5.4.3 Case Study

The proposed algorithm has been applied in a real world business application in Centrelink, Australia. The purpose of the case study is to predict and further prevent debt occurrence based on the customer transactional activity data.

The dataset used for the sequence classification is 1006 J. Comput. Sci. & Technol., Nov. 2009, Vol.24, No.6 composed of customer activity data and debt data. In Centrelink, every customer contact (e.g., a circumstance change) will trigger a sequence of activities. As a result, large volumes of activity-based transactions are recorded in activity transactional files.

In the original activity transactional table, each activity has 35 attributes, of which 4 are used in the case study. These attributes are "CRN" (Customer Reference Number) of a customer, "Activity Code", "Activity Date" and "Activity Time" of each activity respectively shown in Table 5.13. We sort the activity data according to "Activity Date" and "Activity Time" to construct the activity code sequence. The debt data consist of the "CRN" of the debtor and "Debt Transaction Date". In our case study, only the activities of a customer before the occurrence of a debt are kept for the sequence classification task. After data cleaning, there are 15 931 activity sequences including 849 831 activity records used.

Table 5-13 Samples of Centrelink Activity Data

CRN	Act_Code	Act_Date	Act_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	7:38:48
******002	DOC	11/10/2007	8:34:36
******002	RCV	11/10/2007	9:44:39
******002	FRV	11/10/2007	10:18:46
******002	AAI	07/02/2008	15:11:54

In this case study, we build a three-level hierarchical classifier. The thresholds of CCR at the three levels are as follows.

$$T^1 = \begin{cases} 2, & \text{if } CCR > 1, \\ 0.5, & \text{if } CCR < 1, \end{cases}$$
$$T^2 = \begin{cases} 1.2, & \text{if } CCR > 1, \\ 0.8, & \text{if } CCR < 1, \end{cases}$$
$$T^3 = \begin{cases} 1.05, & \text{if } CCR > 1, \\ 0.95, & \text{if } CCR < 1. \end{cases}$$

In order to evaluate the accuracy of the proposed algorithm, we implement another algorithm CCR_{CMAR} which is similar to CMAR[167]. CCR_{CMAR} is also implemented in our hierarchical framework. At each level, the sub-classifier is trained using a weighted χ^2 on multiple patterns. We compare the accuracy of CCR_{CMAR} to our proposed algorithms $CCR_{highest}$ and CCR_{multi} at difference min sup levels. The results are shown in Table 5.14. Note that in all of our experiments, 60% of the dataset is extracted as a training set while the remainder is used as a testing set. Maintaining the ratio of training and testing sets but randomly dividing them, we test the built classifier for five times. In Table 5.14, at all min sup levels, $\text{CCR}_{\text{multi}}$ outperforms $\text{CCR}_{\text{highest}}$. This result again verifies that the classifier constructed from only using the highest ranking pattern for one instance suffers from overfitting. Between the two algorithms both using multiple patterns for one instance, $\text{CCR}_{\text{multi}}$ and CCR_{CMAR}, we can see that $\text{CCR}_{\text{multi}}$ outperforms CCR_{CMAR} at all min sup levels. When min sup becomes greater, the difference between the two algorithms increases, which means our algorithm is more robust than CCR_{CMAR} when less patterns are discovered for classification.

Table 5-14 Performance of Different Algorithms

min_sup (%)	No. Patterns	CCR_{CMAR} (%)	CCR_{highest} (%)	CCR_{multi} (%)
1	39 220	75.0	72.7	75.2
2	10 254	74.4	71.8	74.9
5	1 116	69.4	70.9	72.4
10	208	64.2	61.0	66.7

In order to compare the efficiency of our algorithm and conventional algorithms, we also implement the standard sequential mining algorithm using SPAM[164]. In our case study, SPAM takes too long time if min sup < 5%. So we mined for two sets of sequential patterns, with min_sup = 5% and min_sup = 10%, which are called “PS05” and “PS10”respectively. When min_sup = 5%, the number of the mined patterns is 2,173,691. In the coverage test, we would check whether a pattern covering each sample. Suppose we have 15,931 sequences. The total number of the possible checking between the sequence data and the mined sequential patterns with min sup =5% is 2173691 × 15931 × 0.05 = 1.73 × 109. When min sup = 10%, the number of the mined patterns is 773,724. And the total number of possible matching between the sequence data and the sequential patterns is 773724 × 15931 × 0.1 = 1.12 × 109. Even after pattern pruning, it is still inhibitorily time-consuming to implement serial coverage test and build classifier on such a large set of patterns. In this experiment, we ranked the patterns according to their CCRs and extracted the first 4000 and 8000 patterns from “PS10” and “PS05” and called them “PS10-4K”, “PS05-4K”, “PS10-8K” and “PS05-8K” respectively. Hence we have four classifiers built on the above four sequential pattern sets. The comparison to our classifiers $\text{CCR}_{\text{multi}}$ at min sup = 5% and at min sup = 10% is shown in Table 5.15. In Table 5.15, the “No. Patterns” is the number of the patterns obtained from sequential pattern mining stage. The “Accuracy” is the accuracy of each classifier on the same testing dataset.

Table 5-15 Comparison of the Proposed Algorithm to Conventional Algorithm

min_sup	CCR_{multi}		CCR_{SPAM}			
(%)	No. Patterns	Accuracy (%)	No. Patterns	Accuracy (%)	No. Patterns	Accuracy (%)
10	208	66.7	4 000	64.7	8 000	65.7
5	1 116	72.4	4 000	69.9	8 000	70.6

For the convenience of presentation, we call the algorithm based on SPAM “CCR_{SPAM}”. We have the following two findings from Table 5.15. Firstly, the accuracy of classifier increases when min sup decreases and the number of patterns increases. This finding happens on both our algorithm and CCR_{SPAM}. Actually this finding is also proven by Table 5.14. When the min sup decreases from 10% to 1%, the accuracy of the classifiers will increase monotonically. Secondly, the proposed algorithm outperforms CCR_{SPAM} even though it uses much less patterns than CCRSPAM. When min sup = 10%, there are only 208 patterns mined in our algorithm while the accuracy is 66.7%. Even 8000 patterns are selected for building the sequential classifier in CCR_{SPAM}, the accuracy is 65.7% while the accuracy decreases to 64.7% when input pattern number is 4000. When min sup = 5%, we have the similar finding with a little bigger difference in the classifier accuracy.

From this experiment we can see that our algorithm outperforms CCR_{SPAM} in both efficiency and accuracy. We believe that one of the reasons for the improvement in accuracy is that our algorithm uses less redundant sequential patterns.

6 Mining Combined Social Security Patterns

6.1 Introduction

The notion of association rules [151] was proposed 15 years ago and is widely used today. However, as large numbers of association rules are often produced by association mining, it can sometimes be very difficult for users to not only understand such rules, but also find them a useful source of knowledge to apply to their business processes. Therefore, to present associations in an interesting and effective way, and in order to find actionable knowledge from resultant association rules, a novel idea of combined patterns is proposed. Combined patterns comprise combined association rules, combined rule pairs and combined rule clusters.

A combined association rule is composed of multiple heterogeneous itemsets from different datasets, while combined rule pairs and combined rule clusters are built from combined association rules. The proposed combined patterns provide more interesting knowledge and more actionable results than traditional association rules. The contributions of this paper are: 1) a definition of combined patterns, including combined rules, combined rule pairs and combined rule clusters; 2) interestingness measures designed for combined patterns; 3) two kinds of redundancy (i.e., rule redundancy and rule pair redundancy) identified for combined patterns; and 4) an experimental evaluation of the proposed technique on real-life data.

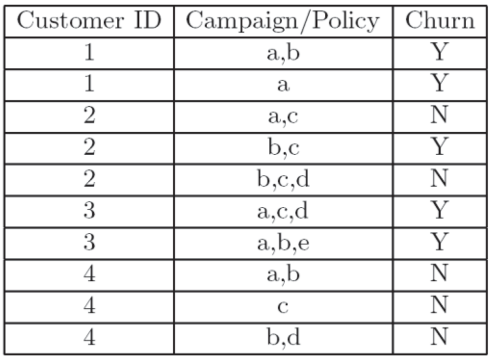
6.2 Related Work

There are often too many association rules discovered from a dataset and it is necessary to conduct post-processing before a user is able to study the rules and identify interesting ones from them. There are many techniques proposed to summarize and/or post-analyze the learned association rules [170,171]. Hilderman et al. proposed to characterize itemsets with information from external databases, e.g., customer or lifestyle data [172]. Their technique works by firstly mining frequent itemsets from transactional data and then partitioning each frequent itemset according to the corresponding characteristic tuple. This method likely results in a large number of rules when many characteristics are involved, with every characteristic having multiple values. Liu and Hsu proposed to rank learned rules by matching against expected patterns provided by user [173]. Rule_Similarity and Rule_Difference are defined to compare the difference between two rules based on their conditions and consequents, and Set_Similarity and Set_Difference are defined to measure the similarity between two sets of rules. The learned rules are ranked by the above similarity/difference and then it is up to the user to identify interesting patterns. In another work, Liu et al. proposed to mine for class association rules and build a classifier based on the rules [168]. With their rule generator, the rule with the highest confidence is chosen from all the rules having the same conditions but different consequents. Liu et al. also proposed direction setting rules to prune and summarize association rules [171]. Chi-square ($\chi 2$) test is used to measure the significance of rules and insignificant ones are pruned. The test is then used again to remove the rules with "expected directions", that is, the rules which are combinations of direction setting rules. Zaiane and Antonie studied strategies for pruning classification rules to build associative classifiers [174]. Their idea selects rules with high accuracy based on the plot of correct/incorrect classification for each rule on the training set. Lent et al. proposed to reduce the number of learned association rules by clustering [170]. Using two-dimensional clustering, rules are clustered by merging numeric items to generate more general rules.

6.3 The Problem

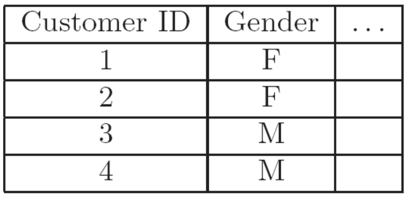
The example that follows illustrates the target problem. Suppose that there are two datasets, transactional dataset and customer demographic dataset (see Tables 6.1 and 6.2), where "Churn" is the behaviour of a customer's switching from a company to another. In the following analysis, campaigns "d" and "e" are ignored to make the result easy to read. The traditional association rules discovered are shown in Table 6.3, and the four rules with lift greater than one are F → Y, M → N, a → Y and c → N. If partitioning the whole population into two groups, male and female, based on the demographic data in Table 6.2, and then mining the two groups separately, some rules are shown in Table 6.4, where Lift_1 and Lift_2 denote respectively the lift of the first/second part of the left side, and I_{rule} is the interestingness of the combined rule. The definitions of the three measures will be given in Section 4.2. We can see from Table 6.4 that more rules with high confidence and lift can be found by combining the rules from two separate datasets.

Table 6-1 Transactional Data



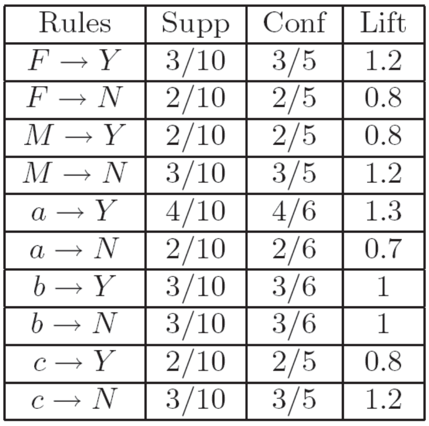
Customer ID	Campaign/Policy	Churn
1	a,b	Y
1	a	Y
2	a,c	N
2	b,c	Y
2	b,c,d	N
3	a,c,d	Y
3	a,b,e	Y
4	a,b	N
4	c	N
4	b,d	N

Table 6-2 Customer Demographic Data



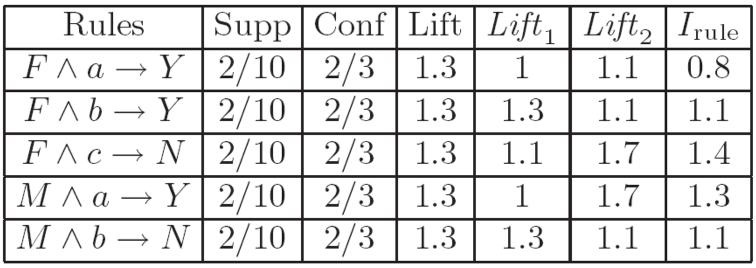
Customer ID	Gender	...
1	F	
2	F	
3	M	
4	M	

Table 6-3 Traditional Association Rules



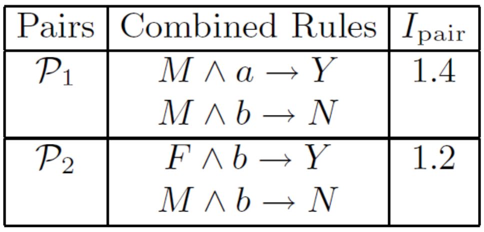
Rules	Supp	Conf	Lift
$F \rightarrow Y$	3/10	3/5	1.2
$F \rightarrow N$	2/10	2/5	0.8
$M \rightarrow Y$	2/10	2/5	0.8
$M \rightarrow N$	3/10	3/5	1.2
$a \rightarrow Y$	4/10	4/6	1.3
$a \rightarrow N$	2/10	2/6	0.7
$b \rightarrow Y$	3/10	3/6	1
$b \rightarrow N$	3/10	3/6	1
$c \rightarrow Y$	2/10	2/5	0.8
$c \rightarrow N$	3/10	3/5	1.2

Table 6-4 Combined Association Rules



Rules	Supp	Conf	Lift	$Lift_1$	$Lift_2$	I_{rule}
$F \wedge a \rightarrow Y$	2/10	2/3	1.3	1	1.1	0.8
$F \wedge b \rightarrow Y$	2/10	2/3	1.3	1.3	1.1	1.1
$F \wedge c \rightarrow N$	2/10	2/3	1.3	1.1	1.7	1.4
$M \wedge a \rightarrow Y$	2/10	2/3	1.3	1	1.7	1.3
$M \wedge b \rightarrow N$	2/10	2/3	1.3	1.3	1.1	1.1

Table 6-5 Combined Rule Pairs



Pairs	Combined Rules	I_{pair}
$\mathcal{P}_1$	$M \wedge a \rightarrow Y$ $M \wedge b \rightarrow N$	1.4
$\mathcal{P}_2$	$F \wedge b \rightarrow Y$ $M \wedge b \rightarrow N$	1.2

Although all the rules in Table 6.4 are of the same confidence and lift, their interestingness values are not the same, which is shown by the last column I_{rule}. For example, for the first rule in Table 6.4, F ∧ a → Y, its interestingness I_{rule} is 0.8, which indicates that the rule is not interesting at all. The explanation is that its lift is the same as the lift of a → Y (see Table 6.3), which means that F contributes nothing in the rule. Therefore, our new measures are more useful than the traditional confidence and lift.

It is more interesting to organize the rules into contrasting pairs shown in Table 6.5, where I_{pair} is the interestingness of the rule pair. P_1 is a rule pair for male group, and it shows that *a* is associated with churn but b with stay. P_1 is actionable in that it suggests b is a preferred action/policy to keep male customers from churning. Moreover, male customers should be excluded when initiating campaign a. P_2 is a rule pair with the same campaign but different demographics. With the same action b, male customers tend to stay, but female tend to churn. It suggests that b is a preferable action for male customers but an undesirable action for female customers.

From the previous example, we can see that rule pairs like P_1 and P_2 provide more information and are more useful and actionable than traditional simple rules shown in Table 6.3 and in this thesis they are referred to as combined patterns. A straightforward way to find the rules in Table 6.4 is to join Tables 6.1 and 6.2 in a pre-processing stage and then apply traditional association rule mining to the derived table. Unfortunately, it is often infeasible to do so in many applications where a dataset contains hundreds of thousands of records or more. Moreover, the rule clusters which organize related rules together are more useful and actionable than individual rules. To find the above useful knowledge like P_1 and P_2, a novel idea of combined patterns will be proposed in the next section.

6.4 Combined Pattern Mining

In this section we provide definitions of combined association rules and combined rule pairs/clusters, and then present their interestingness and redundancy.

6.4.1 Definitions of Combined Patterns

Combined patterns take forms of combined association rules, combined rule pairs and combined rule clusters, which are defined as follows.

Definition 1 (Combined Association Rule). Assume that there are *k* datasets D_i (i = 1..k). Assume I_i to be the set of all items in datasets D_i and $\forall i \neq j$, $I_i \cap I_j = \emptyset$. A combined association rule *R* is in the form of

$$A_1 \wedge A_2 \wedge \ldots \wedge A_k \rightarrow T \qquad (1)$$

where $A_i \subseteq I_i$ (i = 1...k) is an itemset in dataset D_i, $T \neq \emptyset$ is a target item or class and $\exists i, j, i \neq j$, $A_i \neq \emptyset$, $A_j \neq \emptyset$

For example, A_1 can be a demographic itemset, A_2 can be a transactional itemset on marketing campaign, A_3 can be an itemset from a third-party dataset, and T can be the loyalty level of a customer. The combined association rules are then further organized into rule pairs by putting similar but contrasting rules together as follows.

Definition 2 (Combined Rule Pair). Assume that R_1 and R_2 are two combined rules and that their left sides can be split into two parts, U and V, where U and V are respectively itemsets from IU and I_V ($I = \{I_i\}$, $I_U \subset I$, $I_V \subset I$, $I_U \neq \emptyset$, $I_V \neq \emptyset$ and $I_U \cap I_V = \emptyset$). If R_1 and R_2 share a same U but have different V and different right sides, then they build a combined rule pair P as

$$\mathcal{P} : \begin{cases} R_1 : U \wedge V_1 \rightarrow T_1 \\ R_2 : U \wedge V_2 \rightarrow T_2 \end{cases}, \qquad (2)$$

where $U \neq \emptyset$, $V_1 \neq \emptyset$, $V_2 \neq \emptyset$, $T_1 \neq \emptyset$, $T_2 \neq \emptyset$, $U \cap V_1 = \emptyset$, $U \cap V_2 = \emptyset$, $V_1 \cap V_2 = \emptyset$ and $T_1 \cap T_2 = \emptyset$.

A combined rule pair is composed of two contrasting rules, which suggests that for customers with same characteristics U, different policies/campaigns, V_1 and V_2, can result in different outcomes, T_1 and T_2. Based on a combined rule pair, related combined rules can be organized into a cluster to supplement more information to the rule pair.

Definition 3 (Combined Rule Cluster). A combined rule cluster C is a set of combined association rules based on a combined rule pair P, where the rules in C share a same U but have different V in the left side.

$$\mathcal{C}: \begin{cases} U \wedge V_1 \rightarrow T_1 \\ U \wedge V_2 \rightarrow T_2 \\ \dots \\ U \wedge V_n \rightarrow T_n \end{cases}, \tag{3}$$

where U ≠ ∅; ∀i, V_i ≠ ∅, Ti ≠ ∅, U ∩ V_i = ∅; and ∀i ≠ j, $V_i \cap V_j = \emptyset$.

The rules in cluster C have the same U but different V, which makes them associated with various results T. Note that two rules in a cluster may have a same T. For example, assume that there is a rule pair P and a rule cluster C is built based on P by simply adding a third rule as follows.

$$\mathcal{P}: \begin{cases} U \wedge V_1 \rightarrow stay \\ U \wedge V_2 \rightarrow churn \end{cases}, \qquad \mathcal{C}: \begin{cases} U \wedge V_1 \rightarrow stay \\ U \wedge V_2 \rightarrow churn \\ U \wedge V_3 \rightarrow stay \end{cases}. \tag{4}$$

From P, we can see that V_1 is a preferable policy for customers with characteristics U. However, if for some reason, policy V_1 is inapplicable to the specific customer group, P is no longer actionable in that it provides little knowledge on how to prevent the customers from switching to another company. Fortunately, rule cluster C suggests that another policy V_3 can be employed to retain those customers.

6.4.2 Interestingness Measures for Combined Patterns

Traditional interestingness measures contribute little to selecting actionable combined patterns, because they are limited to the traditional simple association rules. Based on traditional supports, confidences and lifts, two new lifts are designed as follows for measuring the interestingness of combined association rules.

$$Lift_U(U \wedge V \to T) = \frac{Conf(U \wedge V \to T)}{Conf(V \to T)} = \frac{Lift(U \wedge V \to T)}{Lift(V \to T)} \tag{5}$$

$$Lift_V(U \wedge V \to T) = \frac{Conf(U \wedge V \to T)}{Conf(U \to T)} = \frac{Lift(U \wedge V \to T)}{Lift(U \to T)} \tag{6}$$

$\mathrm{Lift}_{\mathrm{U}}$(U ∧ V → T) is the lift of U with V as a precondition, which shows how much U contributes to the rule. Similarly, $\mathrm{Lift}_{\mathrm{V}}$ (U ∧ V → T) gives the contribution of V in the rule. Based on the above two new lifts, the interestingness of combined association rules is defined as

$$I_{\text{rule}}(U \wedge V \to T) = \frac{Lift_U(U \wedge V \to T)}{Lift(U \to T)}. \tag{7}$$

$$\begin{aligned} I_{\text{rule}}(U \wedge V \to T) &= \frac{Lift(U \wedge V \to T)}{Lift(U \to T)\; Lift(V \to T)} && (8) \\ &= \frac{Lift_V(U \wedge V \to T)}{Lift(V \to T)}. && (9) \end{aligned}$$

$\mathrm{I}_{\mathrm{rule}}$ indicates whether the contribution of U (or V) to the occurrence of T increases with V (or U) as a precondition. Therefore, "$\mathrm{I}_{\mathrm{rule}} < 1$" suggests that U ∧ V → T is less interesting than U → T and V → T. The value of $\mathrm{I}_{\mathrm{rule}}$ falls in [0,+∞). When $\mathrm{I}_{\mathrm{rule}} > 1$, the higher $\mathrm{I}_{\mathrm{rule}}$ is, the more interesting the rule is.

$\mathrm{I}_{\mathrm{rule}}$ works similarly as direction setting (DS) rules proposed by Liu et al. [171]. The difference is that their method gives a qualitative judgement on a rule whether it is a DS rule or not, while $\mathrm{I}_{\mathrm{rule}}$ is a quantitative measure of the interestingness of a rule. $\mathrm{I}_{\mathrm{rule}}$ measures how much is the unexpectedness of a combined rule against traditional simple association rules.

Interestingness of Combined Rule Pairs and Clusters. Suppose that P is a combined rule pair composed of R_1 and R_2 (See Formula 2), the interestingness of the rule pair P is defined as

$\mathrm{I}_{\mathrm{pair}}(\mathrm{P}) = \mathrm{Lift}_{\mathrm{V}}(\mathrm{R}_1)\ \mathrm{Lift}_{\mathrm{V}}(\mathrm{R}_2)\ \mathrm{dist}(\mathrm{T}_1, \mathrm{T}_2)$ (10)

where dist(·) denotes the dissimilarity between two descendants. It is sometimes written as $I_{pair}(R_1,R_2)$ in this paper. For class with nominal values, such as “Pass” and “Fail”, the dissimilarity can be defined as zero for two same descendants and as 1 for two different descendants. For ordinal class levels, such as “Outstanding, Excellent, Good, Satisfactory, Fail”, the similarity between “Outstanding” and “Fail” can be set to 1 and that between “Excellent” and “Good” can be set to 0.25. I_{pair} measures the contribution of the two different parts in antecedents to the occurrence of different classes in a group of customers with the same demographics or the same transaction patterns. Such knowledge can help to design business campaigns and improve business process. The value of I_{pair} falls in [0,+∞). The larger I_{pair} is, the more interesting a rule pair is.

For a rule cluster C composed of n combined association rules $R_1, R_2, \ldots, R_n$, its interestingness is defined as

$$I_{\text{cluster}}(\mathcal{C}) = \max_{i \neq j, R_i, R_j \in \mathcal{C}, T_i \neq T_j} I_{\text{pair}}(R_i, R_j). \qquad (11)$$

The definition of $I_{cluster}$ that we have provided indicates that interesting clusters are the rule clusters with interesting rule pairs, and the other rules in the cluster provide additional information. Same as I_{pair}, the value of $I_{cluster}$ also falls in [0,+∞).

The interestingness of combined rule pair and cluster is decided by both the interestingness of rules and the most contrasting rules within the pair/cluster. A cluster made of contrasting confident rules is interesting, because it explains why different results occur and what to do to produce an expected result or avoid an undesirable consequence.

Selecting Combined Patterns. With the above interestingness measures, actionable combined patterns will be selected. First, the interesting combined rules are selected from the learned rules with support, confidence, lift, $Lift_U$, $Lift_V$ and I_{rule}. Second, the rules with high support and confidence are organized into pairs and then the pairs are ranked with I_{pair} to find contrasting rule pairs. Finally, related rules are added to selected rule pairs to build rule clusters.

Combined patterns are “actionable” in that: 1) for a single rule, $Lift_v$ measures the contribution of V to the result, which may suggest that V can be used to produce an expected outcome; and 2) the difference in the left hand of contrasting rules within a cluster explains why different results occur and how to get an expected result or avoid an undesirable consequence.

6.4.3 Redundancy in Combined Patterns

There are two kinds of redundancy in combined patterns: 1) the redundancy of combined rules within a rule cluster, and 2) the redundancy of combined rule pairs, which are defined as follows.

Definition 4 (Redundant Combined Association Rule). *Let C be a combined* association rule cluster, and R : U ∧ V → T and R' : U ∧ V' → T' be two combined rules in C, R ∈ C, R'∈ C. R is redundant if V' ⊆ V , T' = T , Lift (R') ≥ Lift (R), Lift_U(R') ≥ Lift_U(R), Lift_V (R') ≥ Lift_V (R) and I_{rule}(R') ≥ I_{rule}(R).

Definition 5 (Redundant Combined Rule Pair). A combined rule pair P is redundant if: 1) there exists a rule pair P' with Ipair(P') ≥ Ipair(P); and 2) for each R : U ∧ V → T ∈ P, there exists a rule R' : U' ∧ V ' → T'∈ P' with U' ⊆ U, V'⊆ V , T'= T , Lift(R') ≥ Lift (R), Lift_U(R') ≥ Lift_U(R), Lift_V (R') ≥ Lift_V (R) and I_{rule}(R') ≥ I_{rule}(R).

Our method for removing the two kinds of redundancy of combined patterns is composed of the following two steps.

1. Removing redundant rules in each rule cluster. This step is similar to the traditional way of removing redundant association rules, but only the redundancy within each rule cluster is removed here. Within each rule cluster C with the same U, each rule R ∈ C is checked to see whether there exist a rule R_ in the same cluster with the same T and greater confidence, Lift, Lift_U, Lift_V and Irule and V'⊆ V. If yes, then R is removed from C as a redundant rule.

2. Pruning redundant rule pairs. This step reduces the number of rule pairs. For two rule pairs P and P', if, for each rule R ∈ P, there exists a rule R'∈ P' with the same T and greater confidence, Lift, Lift_U, Lift_V and I_{rule}, where U' and V' in R' is are respectively subsets of U and V in R, then all the rules in P are redundant with respect to P', and P is a redundant rule pair in terms of P'. So P will be removed to reduce the number of rule pairs.

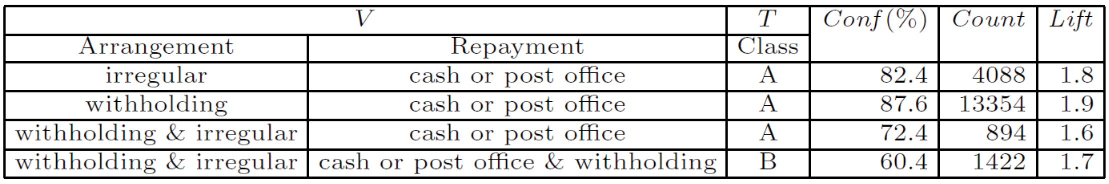
6.5 A Case Study

The technique we propose was tested with real-life data in Centrelink, a Commonwealth Government agency delivering a range of services to the Australian community. The data used was customer debts raised in calendar year 2006 and corresponding customer circumstances data and transactional arrangement / repayment data in the same year. The cleaned sample data contained 355,800 customers and their demographic attributes, as well as individual debt repayment arrangements. The aim was to find the association between demographics, arrangement/repayment methods and the class of customers, which could be used to recover debts as early as possible.

We discovered combined patterns in four steps. Firstly, the transactional data (with arrangements and repayments) was mined for frequent patterns. Secondly, the whole population was partitioned into groups by frequent transactional patterns. Thirdly, the demographic data of each customer group was mined for association rules. And lastly, combined patterns were generated by combining the above results. The minimum support was set to 20 (in the count of customers instead of percentage) and the minimum confidence was set to 60%. To discover interesting combined rules, we set Lift > 1, $Lift_U > 1$, $Lift_V > 1$, $I_{pair} > 1$ and $I_{rule} > 1$, and to discover interesting combined rule clusters, the selected rules were organized into clusters, with the rule clusters then ranked by $I_{cluster}$.

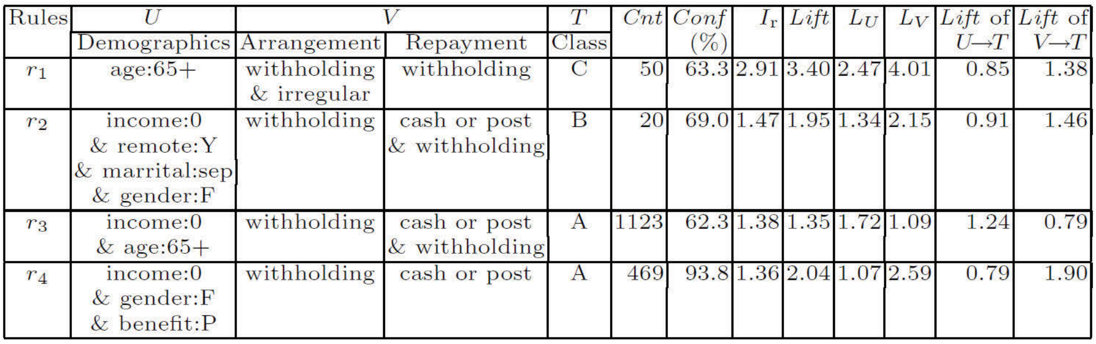
Generally speaking, to prune redundancy in association rules, when two rules have the same confidence and one rule is more general than the other, preference was given to the shorter one. Nevertheless, when analyzing the rules discovered in this exercise, we found that because some rules were on almost the same group of customers, business experts tended to prefer longer rules which provided more detailed information concerning the overall characteristics of the group. Therefore, in this case study, those rules with confidence less than 1.05 times that of more specific rules were removed as redundant rules and the same was done to remove redundant rule clusters.

Table 6-6 Traditional Association Rules



V		*T*	*Conf*(%)	*Count*	*Lift*
Arrangement	Repayment	Class			
irregular	cash or post office	A	82.4	4088	1.8
withholding	cash or post office	A	87.6	13354	1.9
withholding & irregular	cash or post office	A	72.4	894	1.6
withholding & irregular	cash or post office & withholding	B	60.4	1422	1.7

Table 6-7 Selected Combined Rules



Rules	*U*	*V*		*T*	*Cnt*	*Conf*	I_r	*Lift*	L_U	L_V	*Lift* of	*Lift* of
	Demographics	Arrangement	Repayment	Class		(%)					*U*→*T*	*V*→*T*
r_1	age:65+	withholding & irregular	withholding	C	50	63.3	2.91	3.40	2.47	4.01	0.85	1.38
r_2	income:0 & remote:Y & marrital:sep & gender:F	withholding	cash or post & withholding	B	20	69.0	1.47	1.95	1.34	2.15	0.91	1.46
r_3	income:0 & age:65+	withholding	cash or post & withholding	A	1123	62.3	1.38	1.35	1.72	1.09	1.24	0.79
r_4	income:0 & gender:F & benefit:P	withholding	cash or post	A	469	93.8	1.36	2.04	1.07	2.59	0.79	1.90

There were 7,711 association rules before removing redundancy of combined rules. After removing redundancy of combined rules, 2,601 rules were left, which built up 734 combined rule clusters. After removing redundancy of combined rule clusters, 98 rule clusters with 235 rules remained, which was within the capability of human beings to read. The traditional association rules we discovered from transactional data are given in Table 6.6. Some selected combined patterns are shown respectively in Tables 6.7 and 6.8. In the two tables, columns L_U and L_V stand for $Lift_U$ and $Lift_V$, respectively.

In Table 6.7, r_1: "Age:65+, arrangement=withholding and irregular, repayment=withholding → C" has a high I_{rule} of 2.91. "Lift of U → C" indicate that the lifts of "Age:65+ → C" is 0.85, which suggests that "Age:65+" is negatively associated with "C". However, $Lift_U$ = 2.47 suggests that, under "arrangement=withholding and irregular, repayment=withholding", "Age:65+" becomes positively associated with "C". Moreover, $Lift_V$ is greater than "Lift of V → C", which suggests that the contribution of the specific arrangement and repayment to the occurrence of "C" also increases in customer group "Age:65+". What's more, Lift = 3.40 also suggests that the combination of "Age:65+" and "arrangement=withholding and irregular, repayment=withholding" more than triples the probability of the occurrence of "C". Therefore, r_1 is a very interesting rule, which explains why it has a high value of I_{rule}. In contrast, r_5 in Table 6.8 has an I_{rule} of 0.86 (shown as I_r), which indicates that it is not interesting as a single rule. Although r_5 has a high lift of 2.02, its Lift_U and Lift_V are respectively less than “Lift of U → C” and “Lift of V → C”, which suggests that the contribution of U and V to the occurrence of C becomes less when they are combined together. That is, for r_5, $U \wedge V \rightarrow C$ is actually less interesting or useful than U → C and V → C. Nevertheless, it does not necessarily mean that r_5 is not interesting as a part of a rule cluster, since I_{rule} measures the interestingness of a single rule, not that of a rule cluster.

Table 6-8 Selected Combined Rule Clusters

Clu-sters	Ru-les	U	V		T	Cnt	$Conf$ (%)	I_r	I_c	$Lift$	L_U	L_V	$Lift$ of $U{\rightarrow}T$	$Lift$ of $V{\rightarrow}T$
		demographic	arrangement	repayment										
R_1	r_5	age:65+	withhold	cash or post	A	1980	93.3	0.86	6.5	2.02	1.06	1.63	1.24	1.90
	r_6		irregular	cash or post	A	462	88.7	0.87		1.92	1.08	1.55	1.24	1.79
	r_7		withhold & irregular	cash or post	A	132	85.7	0.96		1.86	1.18	1.50	1.24	1.57
	r_8		withhold & irregular	withhold	C	50	63.3	2.91		3.40	2.47	4.01	0.85	1.38
R_2	r_9	marital:sin &gender:F &benefit:N	irregular	cash or post	A	400	83.0	1.12	6.3	1.80	1.01	2.00	0.90	1.79
	r_{10}		withhold	cash or post	A	520	78.4	1.00		1.70	0.89	1.89	0.90	1.90
	r_{11}		withhold & irregular	cash or post & withhold	B	119	80.4	1.21		2.28	1.33	2.06	1.10	1.71
	r_{12}		withhold	cash or post & withhold	B	643	61.2	1.07		1.73	1.19	1.57	1.10	1.46
	r_{13}		withhold & vol. deduct	withhold & direct debit	B	237	60.6	0.97		1.72	1.07	1.55	1.10	1.60
	r_{14}		cash	agent	C	33	60.0	1.12		3.23	1.18	3.07	1.05	2.74
R_3	r_{15}	income:0 &age:22-25	irregular	cash or post	A	191	76.7	1.03	5.1	1.66	0.93	1.85	0.90	1.79
	r_{16}		cash	cash or post	C	440	62.1	1.08		3.34	1.31	2.76	1.21	2.56
R_4	r_{17}	benefit:Y &age:22-25	irregular	cash or post	A	218	79.6	1.15	4.1	1.73	0.97	2.06	0.84	1.79
	r_{18}		cash	cash or post	C	483	65.6	0.78		3.53	1.38	1.99	1.78	2.56

Some selected rule clusters are shown in Table 6.8. The clusters are ordered descendingly by $I_{cluster}$ (shown as I_c). Within each cluster, the rules are ordered first ascending by class and then descending by Lift_V (shown as L_V). For customers with “marital:single, gender:F, benefit:N” (see R_2), “Arrangement= irregular or withholding, Repayment=cash or post office” is associated with class A (see r_9 and r_{10}), while “Arrangement=cash, Repayment=agent recovery” is associated with class C (see r_{14}). Here, Class A is preferable than Class B, and Class B is preferable than Class C. Therefore, for a single female customer with a new debt, if her benefit type is N, she may be encouraged to repay under “Arrangement=irregular or withholding, Repayment=cash or post office”, and be persuaded not to repay under “Arrangement=cash, Repayment=agent recovery”. In such a way, her debt will probably be repaid more quickly. For the above customer group of single female on benefit N, the priority of arrangement-repayment methods is given by the rules from r_9 to r_{14}.

Such kind of knowledge is actionable in that it can help to improve policy or design campaigns to recover debts as soon as possible.

7 Rare Class Association Rule Mining with Multiple Imbalanced Attributes

Problems associated with data imbalance or imbalance data sets are often encountered in data mining – particularly in application-oriented data mining tasks – and applying conventional data mining algorithms to solve these problems is sometimes not all that successful.

For example, in classification rule mining, when a prediction model is trained on an imbalanced data set it may show a strong bias toward the majority class. However, this result is the opposite of what would be expected if the data set was balanced. This problem is called Class Imbalance (Japkowicz 2000) and in recent years it is an area of data mining that has attracted increased attention (Akbani 2004; Zhou 2006; Cao 2008).

The procedure employed to develop a novel algorithm to mine class association rules on datasets with multiple imbalanced attributes includes four steps:

1. Association rules without imbalanced attributes are mined using standard Apriori algorithm (Agrawal, 1994).
2. With respect to one of the imbalanced attributes, the dataset is filtered so that the dominated part is removed.
3. Association rule mining is applied to the filtered dataset, based on new defined measurements.
4. The parameters of rules with imbalanced attributes are transformed so that the rules can be post-processed in a uniform space.

7.1 Background

The problem of data imbalance has attracted increasing interest (Liu 2006; Chawla 2002; Sun 2006 & 2007; Japkowicz 2001) including some specific research related to data imbalance in class association rule mining (Gu et al. 2003; Arunasalam & Chawla 2006; Verhein & Chawla 2007). In particular, Arunasalam and Chawla (2006) presented an algorithm for association rule mining in imbalanced data. Their paper studied the anti-monotonic property of the Complement Class Support (CCS) and applied it into the association rule mining procedure.

Later, Verhein and Chawla (2007) proposed a novel technique, Class Correlation Ratio (CCR), as a measure to deal with data imbalance problems in class association rule mining. However, although their algorithm outperformed earlier algorithms on imbalanced datasets, it still focused on the data imbalance of target class to improve the performance of so-called associative classifiers (Liu, 1998).

In 1999, Liu et al. proposed an algorithm to tackle rare itemset data problems – MSApriori. Because of the similarities between rare itemset problems and data imbalance problems, MSAprior was used in the ensuing years to solve both. In the MSApriori algorithm, the authors refer to minimum item supports (MIS) which have to be assigned to every item by user. Similar to a later algorithm proposed by Yun (2003) this adds considerable time to a task and the performance of the algorithm depends heavily on the specified supports.

7.1.1 Class Association Rules

The algorithm proposed is designed to deal with multiple attribute imbalance problems when mining for class association rule on imbalanced dataset. Notations for class association rule mining are defined below.

Let T be a set of tuples wherein each tuple follows the schema (A_1, A_2, …, A_N, A_C). In this schema (A_1, A_2, …, A_N) are N attributes while A_C is a special attribute – the target class. Collectively, these attributes may be either categorical or continuous.

For continuous attributes, the value range is discretized into intervals. For the convenience of description, an attribute-value pair is referred to as an *item*. Thus, in itemset $U \subseteq A$, A is the itemset of any items with attributes (A_1, A_2, … , A_N), c is an itemset of class attribute, and a class association rule can be represented as $U \Rightarrow c$.

Here, U may contain a single item or multiple items and so the class association rules is represented as $X \cup I \Rightarrow c$, where $X \subseteq A$ is the itemset of balanced attributes and $I \subseteq A$ is the itemset of the imbalanced attributes.

7.1.2 Data Imbalance in Association Rule Mining

A characteristic of data imbalance problems is that they have many more instances of certain attribute values than others. In most applications, the minority parts of an attribute are more interesting than the majority parts. For example, let us say that in a demographic dataset native language speakers far outnumber the people who do not.

However, while the rules consisting of ‘Language: Native’ make common sense, it is the customers who speak other languages that are more interesting to analysts. A further example could be a routine medical examinations dataset where there are many more disease cases than healthy cases. Obviously, the rules consisting of ‘Status: Disease’ are important, but why some people have not succumb to a disease and are ‘Status: Healthy’ is also important to analyze.

Previously, in the majority of association rule mining algorithms, minimum support and minimum confidence are used to select interesting association rules from a large number of frequent patterns. In order to find rules that involve a minority part of an imbalanced attribute, minimum support has to be set very low – and this is likely to result in a large number of uninteresting rules.

7.2 Novel Association Rule Mining Procedure

Today, many algorithms deal with class imbalance problems, however the proposed algorithm does not consider the imbalance problem of the target class, focusing instead on the multiple imbalanced attributes on left-hand side of the class association rules.

In the algorithm, association rule mining is done through two parallel parts – one involving no imbalanced attributes on which a standard *Apriori* algorithm is used to mine interesting rules; the other containing imbalanced attributes that are mined on sub-datasets to achieve high efficiency. My procedure is shown below:

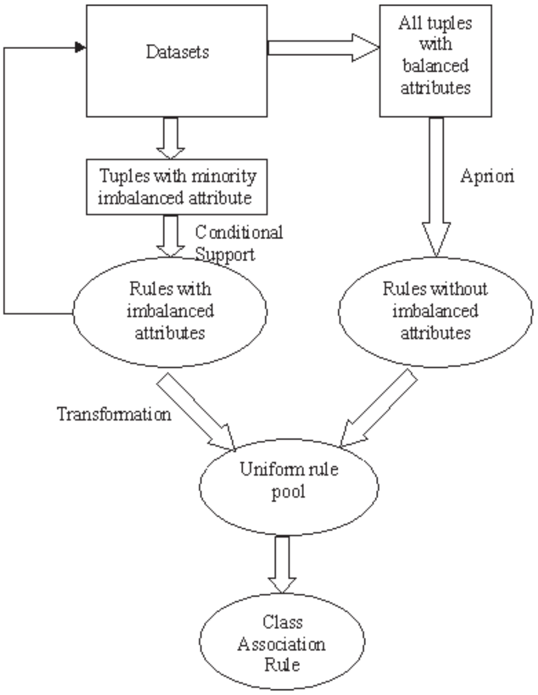

Datasets
All tuples with balanced attributes
Tuples with minority imbalanced attribute
Apriori
Conditional Support
Rules with imbalanced attributes
Rules without imbalanced attributes
Transformation
Uniform rule pool
Class Association Rule

Figure 7-1 Proposed Algorithm

1. Standard association rule mining is applied to the balanced attributes. In the original dataset, the imbalanced attributes are excluded and all of the tuples are kept for association rule mining.

2. The original dataset is filtered to determine the tuples containing minority parts imbalanced attributes – in this step, only a small portion of the dataset is retained.

3. The association rules on the filtered dataset are mined using predefined minimum confidence and minimum conditional supports – for every imbalanced attribute, Step 2 and Step 3 are repeated.

4. Measurements are transformed into a uniform space and all mined rules are combined. The final class association rule list is selected based on a set of criteria.

7.2.1 Interestingness Measures

In standard association rule mining algorithms a number of measurements are used to select interesting rules. For example, the minimum support *minsup*, the minimum confidence *minconf*, the minimum lift *minlift* and so on. In order to mine rules consisting of imbalanced attributes, the definition of support is expanded – since the minority of an imbalanced attribute normally occurs in a small portion of the tuples. That is, *conditional support* is defined to measure the interestingness of the rules with imbalanced attributes. If a class association rule is $X \cup I_m \Rightarrow c$, I_m is the minority part of one imbalanced attribute m, the conditional support of this rule is

$$Supp_c = \frac{P(X \cup I_m \cup c)}{P(I_m)}$$

where X is an itemset of balanced attributes, I_m is a 1-itemset of imbalanced attribute m, c is a class ID.

Suppose the original dataset is represented as T. The subset T_m consists of the tuples containing the minority of imbalanced attribute m. If an association rule is $X \cup I_m \Rightarrow c$, the confidence of this rule is,

$$Conf = \frac{P(X \cup I_m \cup c)}{P(X \cup I_m)}$$

The expected confidence is

$$Conf_E = P(c)$$

and the lift is

$$Lift = \frac{Conf}{Conf_E} = \frac{\frac{P(X \cup I_m \cup c)}{P(X \cup I_m)}}{P(c)} = \frac{P(X \cup I_m \cup c)}{P(X \cup I_m) \cdot P(c)}$$

When an association rule $X \cup I_m \Rightarrow c$ is mined on the dataset T_m, the conditional support is

$$Supp'_c = P'(X \cup I_m \cup c) = \frac{P(I_m \cup (X \cup I_m \cup c))}{P(I_m)}$$

because T_m only has the tuples containing minority of the imbalanced itemset I_m,

$$P(I_m \cup (X \cup I_m \cup c)) = P(X \cup I_m \cup c)$$

Hence,

$$Supp'_c = P'(X \cup I_m \cup c) = \frac{P(I_m \cup (X \cup I_m \cup c))}{P(I_m)} = Supp_c$$

Similarly, the confidence on the sub-dataset is

$$Conf = \frac{P'(X \cup I_m \cup c)}{P'(X \cup I_m)} = \frac{\frac{P(I_m \cup (X \cup I_m \cup c))}{P(I_m)}}{\frac{P(I_m \cup (X \cup I_m))}{P(I_m)}} = \frac{P(X \cup I_m \cup c)}{P(X \cup I_m)} = Conf$$

Obviously, either on the original dataset *T* or filtered dataset Tm, the conditional support and confidence of the mined rule keep invariant. However, on filtered dataset T_m, the expected confidence is

$$Conf'_E = P'(c) = \frac{P(I_m \cup c)}{P(I_m)}$$

So the lift on the filtered dataset T_m is

$$Lift' = \frac{Conf'}{Conf'_E} = \frac{\frac{P'(X \cup I_m \cup c)}{P'(X \cup I_m)}}{P'(c)} = \frac{P(X \cup I_m \cup c) \cdot P(I_m)}{P(X \cup I_m) \cdot P(I_m \cup c)} \neq lift$$

7.2.2 Transformation

As the algorithm undertakes association rule mining in two parallel parts, we firstly transform these into a uniform space so that as few as possible measurements are defined by user. From the above analysis, the conditional support and confidence of the rules is same for either on original dataset or filtered dataset. Thus, in order to use uniform criteria to select the rules, the lift on the filtered dataset has to be transformed.

In original dataset *T*, the expected confidence with respect to *c* is *P(c)*. Based on the confidence for filtered dataset T_m, the lift obtained from filtered dataset is as follows

$$L_{new} = \frac{Conf'}{Conf_E} = \frac{Conf'}{P(c)}$$

As the confidence, lift and the conditional support of all the rules have the same base, a uniform criterion is then applicable to select the final rules.

7.3 Test Case

The algorithm is tested to mine an association rule based on the demographic attributes, debt information and repayment arrangement for customers who owed a debt to the Commonwealth. The target class(es) is whether a customer was a 'fast re-payer', 'moderate re-payer', or 'slow re-payer; the idea being that this information would assist Centrelink in putting in place suitable repayment arrangements.

The combined pattern mining proposed by Zhao (2007) is used here, which is defined as

$$\begin{cases} X \cup Y_1 \Rightarrow c_1 \\ \qquad \vdots \\ X \cup Y_i \Rightarrow c_j \end{cases},$$

where c_j is class ID and Y_i is the itemset of actionable attributes.

In this case study, Y_i is an arrangement pattern of a customer. From the above equation we can see that each association rule in the combined pattern is a class association rule. Hence we can apply the proposed algorithm to mine the class association rule in the combined pattern.

7.3.1 Datasets Involved

Three datasets are used: customer demographic data, debt data and repayment data. The first dataset contained demographic circumstances of customers, such as customer ID, gender, age, marital status, number of children, income, location, language, prefer language and so on. The second data contained debt related information, such as the date of debt raised, the amount of debt, the outstanding balance of the debt, and the benefit type related to the debt. While the repayment dataset included the debt repayment amount, the debt re-repayment date, the type of the repayment and the type of debt recovery arrangement – which is an agreement between a customer and Centrelink concerning the method, amount and frequency of repayments. The class IDs, which were defined by business experts, were included in the repayment dataset.

The data used were debts raised in calendar year 2006 and after removing noises such as repayments with zero or negative amounts there were 479,288 customers in the demographic dataset and 2,627,348 repayments in repayment dataset.

The attributes in the debt and repayment datasets were relatively balanced. However, in the demographic datasets, three attributes had imbalanced distributions – 'Remote' (referring to place of residence in remote locations across Australia), 'Lang' (principal language spoken by the customer) and 'Indig' (whether a customer identifies as being Aboriginal or of Aboriginal decent). The distribution of these three attributes is shown in Fig. 7.2.

7.3.2 Experimental Results

In our experiment, customers are grouped based on Arrangement Patterns. Selected association rules on balanced attributes are shown in Table 7.1, while selected rules with imbalanced attributes are shown in Table 7.2 on the following page:

Table 7-1 Selected Results with Balanced Attributes

Arrangement	Demographic Pattern	Class	*Conf$_E$(%)*	*Conf(%)*	*Supp(%)*	*Lift*	*Count*
C_A	Marrital:SIN & Gender:F & Benefit:AAA	Slow Payer	51	60	6.4	1.2	33
CI_W	Benefit:BBB	Quick Payer	40.8	67	4.9	1.6	61
W_W	Weekly:0 & age:65y+	Quick Payer	72.4	88.7	8.9	1.2	110
WV_WC	Benefit:AAA	Quick Payer	72.4	88.4	8.7	1.2	107
V_V	Weekly:0 & Gender:M	Quick Payer	72.4	86	9	1.2	111
CI_W	Marrital:SIN & Gender:F & Benefit:BBB	Moderate Payer	60.4	80.4	5.1	1.3	119
WI_W	Weekly:[$400, $600) & Marrital:SEP & age:26y-50y	Moderate Payer	56.6	65.4	2.6	1.2	100

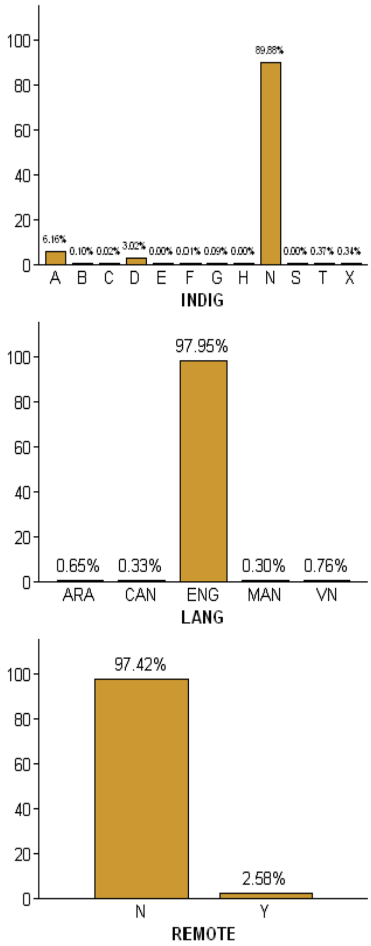


Figure 7-2 Distribution of the Imbalanced Attributes

Table 7-2 Selected Rules with Imbalanced Attributes Caption Style

Arrangement	Demographic Pattern	Class	$Conf_E(\%)$	$Conf(\%)$	$Supp_c(\%)$	L_{new}	*Count*
W_W	Weekly:[$200 $400) & INDIG:A &GENDER:F	Moderate Payer	39	48.6	6.7	1.2	52
C_A	MARRITAL:SEP & INDIG:A	Slow Payer	25.6	63.3	6.4	2.5	50
CI_A	Weekly:[$400 $600) & INDIG:D	Quick Payer	35.4	64.9	6.4	1.8	50
WV_W	MARRITAL:SEP & INDIG:D & Children:0	Slow Payer	39	49.8	16.3	1.3	127
V_V	Weekly:0 & MARITAL:MAR & LANG:ARA	Moderate Payer	25.6	46.9	7.8	1.8	61
WV_WV	LANG:MAN & GENDER:F	Quick Payer	25.6	49.7	11.4	1.9	89
WI_CW	Weekly:[$200 $400) & REMOTE:Y & GENDER:F	Quick Payer	39	45.7	18.8	1.2	147

The lift of each of the rules is then transformed so that they could be measured using uniform selection criteria, and from the resultant pool mined the final association rules shown below:

Table 7-3 Selected Results of the Combined Association Rules

Arrangement	Demographic Pattern	Class
C_A	Marrital:SEP & Gender:F & Benefit:AAA	Slow Payer
V_V	Marrital:SEP & Gender:F & Benefit:AAA	Quick Payer
W_A	Marrital:SEP & Gender:F & Benefit:AAA	Moderate Payer
W_W	Marrital:SEP & Gender:F & Benefit:AAA	Slow Payer
WI_CW	GENDER:F & Children:0 & Age:26y-50y	Moderate Payer
CI_C	GENDER:F & Children:0 & Age:26y-50y	Quick Payer
W_A	GENDER:F & Children:0 & Age:26y-50y	Slow Payer
WV_V	GENDER:F & Children:0 & Age:26y-50y	Moderate Payer
C_A	GENDER:F & Children:0 & Age:26y-50y	Slow Payer
WI_CW	Weekly:[$400, $600) & INDIG:D	Slow Payer
C_A	Weekly:[$400, $600) & INDIG:D	Quick Payer
WV_V	Weekly:[$400, $600) & INDIG:D	Quick Payer
CI_CW	Weekly:[$400, $600) & INDIG:D	Moderate Payer
WI_C	LANG:ARA & GENDER:F	Slow Payer
CI_C	LANG:ARA & GENDER:F	Moderate Payer
WI_C	Weekly:[$200, $400) & REMOTE:Y & GENDER:F	Quick Payer
C_A	Weekly:[$200, $400) & REMOTE:Y & GENDER:F	Slow Payer
WV_V	Weekly:[$200, $400) & REMOTE:Y & GENDER:F	Slow Payer
WI_W	Weekly:[$200, $400) & REMOTE:Y & GENDER:F	Moderate Payer

7.4 Conclusions

Unlike algorithms dealing with class imbalance, the proposed algorithm proposes an efficient way to mine class association rules on datasets with multiple imbalanced attributes. Also different from the algorithms dealing with rare item problem, the algorithm employs uniform selection criteria to discover final combined association rules.

8 Conclusions and Future Work

8.1 Conclusions

With the occurrence of the global financial crisis, more and more governments have realized the necessity of enhancing social security services objectives and quality. Data mining and machine learning can play a critical role, as we have demonstrated in mining Australian social security data for debt prevention, recovery, customer analysis, etc., during the past few years. However, as the literature review shows, mining social security (and public sector) data are still an open field for business applications in data mining and machine learning. Very few references have been publicized. In this work, for the first time in the community, we present a picture of studies on social security issues and summarize the key concepts, goals, tasks, and challenges of SSDM, based on our experience and knowledge accumulated through conducting data mining in Australian social security data.

We have also highlighted several case studies of mining social security data, including modelling the impact of activity/activity sequences, mining impact-targeted activity patterns, mining positive and negative sequential patterns, conducting impact-targeted sequence classification, and mining combined association rules. We have discussed how the identified patterns are converted into knowledge that can support business people in a more user-friendly way to take decision-making actions.

8.2 Future Works

8.2.1 Data Mining Applications in Social Security

Firstly, given the likelihood that hundreds or possibly thousands of rules are identified after pruning redundant patterns, how can we efficiently select interesting patterns from them? Secondly, how can domain knowledge be effectively incorporated in data mining procedure to reduce the search space and running time of data mining algorithms? Thirdly, given that the business data is complicated and a single debt activity may be linked to several customers, how can existing approaches for sequence mining be improved to take into consideration the linkage and interaction between activity sequences?

And lastly and perhaps most importantly, how can these discovered rules be used to build an efficient debt prevention system to effectively detect debt in advance and give appropriate suggestions to reduce or prevent debt? The above will be part of our future work.

8.2.2 Sequential Rules Mining and Sequence Classification

In mining negative sequential rules, it can only deal with a single event on the right side. Sometimes it may be interesting to find more generalized negative sequential rules, which will be included in our future work. Moreover, negative sequential rules with time constraints will also be a part of our future research.

In our current adaptive framework of sequence classification, it refines the classifier round by round, and in each round the adaptation is based on the classifier generated in the last round. Though it tracks the evolvement of sequential patterns, the latest pattern variation is given the same consideration as previous ones. In our future work, we will study how to apply tilted weight to the historical data, which may involve including later sequence pattern characteristics into the classification model.

8.2.3 Development of Further Models

In addition to RA, AGE and NSA mining activities of the research section to date include developing predictive models for:

- All earned income reviews;
    - PAYG (pay as you go reviews)
    - TDF (tax file declaration reviews)
    - DMP (data matching reviews)
    - DEEWR (all Department of Education, Employment and Work Place Relations reviews);
- Carer Allowance;
- Carer Payment; and
- Disability Support payment (DSP) reviews

In total, this work provides a risk-rating for over two thirds of Centrelink's 6.5 million customer base, enabling staff to implement prevention measures for those customers at risk of being on an incorrect payment.

8.2.4 A Straightforward Approach to Ongoing Research and Development

As a newly established team we are committed to ensuring that our ongoing research remains methodologically rigorous and focused on supporting the Government of the day in its challenge to deliver a difficult Service Delivery Reform agenda.

The essential need is to bridge the critical gap between 'knowledge discovered' and 'knowledge applied' through adhering to a knowledge transfer process based on social interaction remains at the forefront of our planning - because few would disagree that information and knowledge are most valuable when they are put to use.

The challenge for us is to integrate the resources and knowledge held within Centrelink; while the key to forging the link between knowledge and practice lies in keeping our colleagues informed from beginning to end, providing opportunities for them to evaluate and discuss the information we send to them, and finally, assist them in the process of applying new knowledge into to their workplace practices.

> *"If to do were as easy as to know what is good to do, then chapels would be churches and poor men's cottages palaces."*
>
> *Shakespeare*

List of Publications

Awards

L. Cao, H. M. Bohlscheid, Y. Zhao, H. Zhang, P. Newbigin, B. Clark, Y. Ou, J. Li, Y. Yang, C. Zhang, and Y. Xiao, “Social Security Data Mining for Public Services”, In Top Ten Data Mining Case Studies, ICDM 2010, Australia, 2010.

Book Chapters

Y. Zhao, H. Zhang, L. Cao, H. Bohlscheid, Y. Ou, and C. Zhang, “Data mining applications in social security,” in Data Mining for Business Applications”, L. Cao, P. S. Yu, C. Zhang, and H. Zhang, Eds. New York: Springer, pp. 81–96, 2009.

H. Zhang, Y. Zhao, L. Cao, C. Zhang and H. Bohlscheid, “Rare Association Rule Mining and Knowledge Discovery: Technologies for Infrequent and Critical Event Detection”, ISBN: 978-1-60566-754-6, Information Science Reference, pp. 66-75, 2009.

Conference Papers

Y. Zhao, H. Zhang, L. Cao, C. Zhang, and H. Bohlscheid, “Combined pattern mining: From learned rules to actionable knowledge,” in Advances in Artificial Intelligence. New York: Springer-Verlag, pp. 393–403, 2008.

Y. Zhao, H. Zhang, L. Cao, C. Zhang, and H. Bohlscheid, “Efficient mining of event-oriented negative sequential rules,” in Proc. IEEE/WIC/ACM Int. Conf. Web Intell., pp. 336–342, 2008.

S. Wu, Y. Zhao, H. Zhang, C. Zhang, L. Cao, and H. Bohlscheid, “Debt detection in social security by adaptive sequence classification,” in Proc. 3rd Int.Conf., Knowl. Sci., Eng.Manage. New York: Springer-Verlag, pp. 192–203, 2009.

Y. Zhao, H. Zhang, S. Wu, J. Pei, L. Cao, C. Zhang, and H. Bohlscheid, “Debt detection in social security by sequence classification using both positive and negative patterns,” in

Machine Learning and Knowledge Discovery in Databases. New York: Springer-Verlag, pp. 648–663, 2009.

Y. Zhao, H. Zhang, L. Cao, C. Zhang, and H. Bohlscheid, "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'09), Bangkok, Thailand, pp. 656-663, 2009.

Y. Zhao, H. Bohlscheid, S. Wu and L. Cao, "Less Effort, More Outcomes: Optimising Debt Recovery with Decision Trees", in IEEE International Conference on Data Mining Workshops (ICDMW), pp. 655 -660, 2010.

Journal Articles

H. Zhang, Y. Zhao, L. Cao, C. Zhang, and H. Bohlscheid, "Customer activity sequence classification for debt prevention in social security," J. Comput. Sci. Technol., vol. 24, no. 6, pp. 1000–1009, 2009.

References

[1] H. Aaron, “Demographic effects on the equity of social security benefits,” in The Economics of Public Services, M. Feldstein and R. Inman, Eds., London: Macmillan, 2007.

[2] B. Agarwal, “Social security and the family: Coping with seasonality and calamity in rural India,” J. Peasant Stud., vol. 17, pp. 341–412, 1990.

[3] L. Alexander and T. Jabine, “Access to social security microdata files for research and statistical purposes,” Soc. Secur. Bull., vol. 41, no. 8, pp. 3–17, 1978.

[4] A. J. Auerbach and L. J. Kotlikoff, “An examination of empirical tests of social security and savings,” National Bureau of Economic Research, Cambridge, MA, Working Paper 730. [Online]. Available: http://www.nber.org/papers/w0730, 1984.

[5] A. J. Auerbach and L. J. Kotlikoff, “Simulating alternative social security responses to the demographic transition,”National Bureau of Economic Research, Cambridge,MA,Working Paper 1308. [Online]. Available: http://ideas.repec.org/p/nbr/nberwo/1308.html, 1985.

[6] D. Baker and M.Weisbrot, “Social Security: The Phony Crisis. Chicago”, IL: Univ. of Chicago Press, 2000.

[7] B. D. Bernheim, “Social security benefits: An empirical study of expectations and realizations,” National Bureau of Economic Research, Cambridge, MA, Working Paper 2257. [Online]. Available: http://ideas.repec.org/p/nbr/nberwo/2257.html, 1987.

[8] R. J. Barro and C. Sahasakul, “Average marginal tax rates from social security and the individual income tax,” J. Bus., vol. 59, no. 4, pp. 555–566, 1986.

[9] H. Berghel, “Identity theft, social security numbers, and the web,” Commun. ACM, vol. 43, no. 2, pp. 17–21, 2000.

[10] D. Blanchet and L. P. Pele, "Social security and retirement in France," National Bureau of Economic Research, Cambridge, MA, Working Paper 6214. [Online]. Available: http://ideas.repec.org/p/nbr/ nberwo/6214.html, 1987.

[11] D. Bloom, D. Canning, R. Mansfield, and M. J. Moore, "Demographic change, social security systems, and savings," National Bureau of Economic Research, Cambridge, MA, Working Paper 12621. [Online]. Available: http://econpapers.repec.org/RePEc:nbr:nberwo:12621, 2006.

[12] R. W. Boadway and D. E. Wildasin, "A median voter model of social security," Int. Econom. Rev., vol. 30, no. 2, pp. 307–328, 1989.

[13] M. J. Boskin and G. F. Break, "The Crisis in Social Security: Problems and Prospects" Oakland, CA: Inst. Contemporary Stud., 1977.

[14] G. Burtless and R. A. Moffitt, "Social security, earnings tests, and age at retirement," Public Finance Rev., vol. 14, no. 1, pp. 3–27, 1986.

[15] L. Cao, "In-depth behaviour understanding and use: The behaviour informatics approach," Inf. Sci., 180, no. 17, pp. 3067–3085, 2010.

[16] L. Cao. "Social Security and Social Welfare Data Mining: An Overview", IEEE Trans. SMC Part C, 42(6): 837-853, 2012.

[17] L. Cao, D. Luo, and C. Zhang, "Knowledge actionability: Satisfying technical and business interestingness," Int. J. Bus. Intell. Data Mining, vol. 2, no. 4, pp. 496–514, 2007.

[18] L. Cao, Y. Ou, and P. S. Yu. (2011). "Coupled behaviour analysis with applications," IEEE Trans. Knowl. Data Eng., Please note: to be published.

[19] L. Cao, Y. Ou, P.S. Yu, and G. Wei, "Detecting abnormal coupled sequences and sequence changes in group based manipulative trading behaviors," in Proc. Knowl. Discovery Data Mining, pp. 85–94, 2010.

[20] L. Cao, P. S. Yu, C. Zhang, and Y. Zhao, "Domain Driven Data Mining" New York: Springer-Verlag, 2009.

[21] L. Cao, P. S. Yu, C. Zhang, and H. Zhang, "Data Mining for Business Applications" New York: Springer-Verlag, 2008.

[22] L. Cao, H. Zhang, Y. Zhao, D. Luo, and C. Zhang, "Combined mining: Discovering informative knowledge in complex data," IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 41, no. 3, pp. 699–712, 2011.

[23] L. Cao, Y. Zhao, and C. Zhang, "Mining impact-targeted activity patterns in imbalanced data," IEEE Trans. Knowl. Data Eng., vol. 20, no. 8, pp. 1053–1066, 2008.

[24] L. Cao, Y. Zhao, C. Zhang, and H. Zhang, "Activity mining: From activities to actions," Int. J. Inf. Technol. Decis. Making (IJITDM), vol. 7, no. 2, pp. 259–273, 2008.

[25] L. Cao, Y. Zhao, F. Figueiredo, Y. Ou, and D. Luo, "Mining high impact exceptional behaviour patterns," in Proc. Int. Workshops Emerg. Technol. Knowl. Discovery Data Mining, pp. 56–63, 2007.

[26] L. Cao, Y. Zhao, H. Zhang, D. Luo, C. Zhang, and E. K. Park, "Flexible frameworks for actionable knowledge discovery," IEEE Trans. Knowl. Data Eng., vol. 22, no. 9, pp. 1299–1312, 2009.

[27] R. Carr-Hill, J. Jamison, D. O'Reilly,M. Stevenson, J. Reid, and B. Merriman, "Risk adjustment for hospital use using social security data: Cross sectional small area analysis," Brit. Med. J., vol. 324, p. 390, 2002.

[28] C. Coile, P. Diamond, J. Gruber, and A. Jousten, "Delays in claiming social security benefits," J. Public Econ., vol. 84, no. 3, pp. 357–385, 2002.

[29] H. Cronqvist and R. H. Thaler, "Design choices in privatized socialsecurity systems: Learning from the Swedish experience," Amer. Econ. Rev., vol. 94, no. 2, pp. 424–428, 2004.

[30] P. Cutright, "Political structure, economic development, and national social security programs," Amer. J. Sociol., vol. 70, no. 5, pp. 537–550, 1965.

[31] M. R. Darby, "The effects of social security on income and the capital stock," UCLA Dept. of Econ., Los Angeles, UCLA Econ. Working Paper 095, 1978.

[32] H. Dean and M. Melrose, “Manageable discord: Fraud and resistance in the social security system,” Soc. Policy Administ., vol. 31, no. 2, pp. 103–118, 1997.

[33] P. A. Diamond and P. R. Orszag, “Saving Social Security: A Balanced Approach” Washington, DC: Brookings Institution, 2005.

[34] P. A. Diamond, “Taxation, Incomplete Markets, and Social Security” Cambridge, MA: The MIT Press, 2003.

[35] P. A. Diamond, “A framework for social security analysis,” J. Public Econ., vol. 8, no. 3, pp. 275–298, 1977.

[36] X. Dong, Z. Zhao, L. Cao, Y. Zhao, C. Zhang, J. Li, W. Wei, and Y. Ou, “e-NSP: Efficient negative sequential pattern mining based on identified positive patterns without database rescanning,” in Proc. Conf. Inf. Knowl. Manage., pp. 825–830, 2011.

[37] D. Duncan, H. C. M. Kum, K. Flair, and W.Wang, “Successfully adopting IT for social welfare program management,” in Proc. Annu. Nat.Conf. Digital Govern. Res., pp. 1–9, 2004.

[38] D. Duncan, H. Kum, K. Flair, J. Stewart, E. Weigensberg and P. Lanier, “NC child welfare program” [Online]. Available: http://ssw.unc.edu/ma/index.html, 2007.

[39] D. F. Duncan, H.-C. Kum, E. C.Weigensberg, K. A. Flair, and C. J. Stewart, “Informing child welfare policy and practice using knowledge discovery and data mining technology via a dynamic web site,” Child Maltreat, vol. 13, no. 4, pp. 383–391, 2008.

[40] C. A. Echevarr´ıa and A. Iza, “Life expectancy, human capital, social security and growth,” J. Public Econ., vol. 90, no. 12, pp. 2323–2349, 2006.

[41] Z. Eckstein, M. Eichenbaum, and D. Peled, “Uncertain lifetimes and the welfare enhancing properties of annuity markets and social security,” J. Public Econ., vol. 26, no. 3, pp. 303–326, 1985.

[42] M. Gonzalez-Eiras, “Social security as Markov equilibrium in OLG models: A note,” Rev. Econ. Dyn., vol. 14, no. 3, pp. 549–552, 2011.

[43] M. Feldstein, “The future of social security pensions in Europe,” National Bureau of Economic Research, Cambridge, MA, Working Paper 8487, 2001.

[44] V. Galasso and P. Profeta, “The political economy of social security: A survey,” Eur. J. Political Econ., vol. 18, no. 1, pp. 1–29, 2002.

[45] D. M. Garrett, “The effects of differential mortality rates on the progressivity of social security,” Econ. Inquiry, vol. 33, pp. 457–475, 1995. Please note: This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination. 16 IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS.

[46] C. Gillion, “Social security pensions: development and reform,” Geneva, International Labour Office, 2000.

[47] M. S. Gordon, “Social Security Policies in Industrial Countries; A Comparative Analysis” Cambridge, U.K.: Cambridge Univ. Press, 2009.

[48] E. M. Gramlich, “Different approaches for dealing with social security,” Amer. Econ. Rev., vol. 86, no. 2, pp. 358–362, 1996.

[49] J. Gruber and P. Orszag, “What to do about the social security earnings test?” Centre for Retirement Research, Issues in Brief ib-1. [Online]. Available: http://ideas.repec.org/p/crr/issbrf/ib-1.html, 2003.

[50] J. Gruber and D. A. Wise, “Social security programs and retirement around the world,” Res. Labor Econ., vol. 18, pp. 1–40, 1999.

[51] A. L. Gustman and T. L. Steinmeier, “The social security retirement earnings test, retirement and benefit claiming,” National Bureau of Economic Research, Cambridge, MA, Working Paper 10905, Nov. 2004.

[52] D. Guest, “The Emergence of Social Security in Canada”, 3rd ed. ed. Vancouver, BC, Canada: UBC Press, 1997.

[53] M. Feldstein, “The optimal level of social security benefits,” National Bureau of Economic Research, Cambridge, MA, Working Paper 0970, Aug. 1986.

[54] L. Friedberg, “The labor supply effects of the social security earnings test,” Rev. Econ. Statist., vol. 82, no. 1, pp. 48–63, 2000.

[55] M. Feldstein, “The missing piece in policy analysis: Social security reform,” Amer. Econ. Rev., vol. 86, no. 2, pp. 1–14, 1996.

[56] M. S. Feldstein and J. B. Liebman, Eds., “The Distributional Aspects of Social Security and Social Security Reform,” Chicago, IL: Univ. of Chicago Press, 2002.

[57] S. J. Haider and G. Solon, “Nonrandom selection in the HRS social security earnings sample,” RAND—Labor and Population Program, Working Papers. [Online]. Available: http://econpapers.repec.org/ RePEc:fth:randlp:00-01, 2000.

[58] P. Henman and M. Adler, “Information technology and the governance of social security,” [Online]. Available: http://espace.library.uq.edu.au/view/ UQ:166078, 2003.

[59] A. Hicks, “Qualitative comparative analysis and analytical induction: The case of the emergence of the social security state,” Sociol. Methods Res., vol. 23, no. 1, pp. 86–113, 1994.

[60] M. Hill, “Social Security Policy in Britain” Cheltenham, U.K.: Edward Elgar, 1990.

[61] H. Huang, S. Imrohoroglu, and T. J. Sargent, “Two computations to fund social security,” Macroecon. Dyn., vol. 1, no. 1, pp. 7–44, 1997.

[62] M. D. Hurd, J. P. Smith, and J. M. Zissimopoulos, “The effects of subjective survival on retirement and social security claiming,” J. Appl. Econometr., vol. 19, pp. 761–775, 2004.

[63] A. Imrohoroglu, S. Imrohoroglu, and D. H. Joines, “Computing models of social security,” QM&RBC Codes, Quantitat. Macroecon. Real Bus. Cycles, 1998.

[64] J. Kim, “The impact of e-government on child support enforcement policy outcomes,” in Proc. 8th Annu. Int. Conf. Digital Govern. Res., pp. 212–221, 2007.

[65] W. van der Klaauwa and K. I. Wolpinb, “Social security and the retirement and savings behaviour of low-income households,” J. Econometr., vol. 145, pp. 21–42, 2008.

[66] L. J. Kotlikoff, K. Smetters, and J. Walliser, “Distributional effects in a general equilibrium analysis of social security,” in The Distributional Aspects of Social Security and Social Security Reform (ser. NBER Chapters). Cambridge, MA: Nat. Bureau Econ. Res., pp. 327–370, 2002.

[67] A. B. Krueger and J.-S. Pischke, “The effect of social security on labor supply: A cohort analysis of the notch generation,” J. Labor Econ., vol. 10, no. 4, pp. 412–437, 1992.

[68] H.-C. M. Kum, D. Duncan, K. Flair, and W. Wang, “Social welfare program administration and evaluation and policy analysis using knowledge discovery and data mining (KDD) on administrative data,” in Proc. Annu. Nat. Conf. Digital Govern. Res., pp. 1–6, 2003.

[69] H.-C. M. Kum, D. Duncan, and W.Wang, “Understanding social welfare service patterns using sequential analysis,” in Proc. Annu. Nat. Conf. Digital Govern. Res., pp. 1–2, 2004.

[70] C. Mesa-Lago, “Social security in Latin America,” in Latin Amer. Res. Rev., 2007.

[71] L. J. Kotlikoff, K. A. Smetters, and J. Walliser, “Social security: Privatization and progressivity,” National Bureau of Economic Research, Cambridge, MA, Working Paper 6428, 1998.

[72] R. Lee and S. Tuljapurkar, “Stochastic forecasts for social security,” in Frontiers in the Economics of Aging (ser. NBER Chapters). Cambridge, MA: Nat. Bureau Econ. Res., pp. 393–428, 1998.

[73] D. R. Leimer and S. D. Lesnoy, “Social security and private saving: New time-series evidence,” J. Political Econ., vol. 90, no. 3, pp. 606–629, 1982.

[74] D. R. Leimer, “Lifetime redistribution under the social security program: A literature synopsis,” Soc. Secur. Bull., vol. 62, no. 2, pp. 43–51, 1999.

[75] J. C. Leung, “Social security reforms in China: Issues and prospects,” Int. J. Soc. Welfare, vol. 12, pp. 73–85, 2003.

[76] T. R. Marmor and J. L. Mashaw, “Understanding social insurance: Fairness, affordability, and the modernization of social security and medicare,” Health Affairs, vol. 25, no. 3, pp. 114–134, 2006.

[77] D. McAullay, G.Williams, J. Chen, H. Jin, H. He, R. Sparks, and C. Kelman, “A delivery framework for health data mining and analytics,” in Proc. 28th Australasian Conf. Comput. Sci., pp. 381–387, 2005.

[78] J. Millar, “Understanding Social Security: Issues for Policy and Practice”, 2nd ed. ed. Bristol, U.K.: Policy Press, 2009.

[79] O. S. Mitchell and J. W. Phillips, “Retirement responses to early social security benefit reductions,” National Bureau of Economic Research, Cambridge, MA, Working Paper 7963, 2000.

[80] O. S. Mitchell and J. W. Phillips, “Social security replacement rates for alternative earnings benchmarks,” Retirement Research Center, Univ. Michigan, Ann Arbor, Working Paper wp116, 2006.

[81] J. A. Olson, “Linkages with data from social security administrative records in the health and retirement study,” Social Security Bulletin, 1999.

[82] E. Ooghe, E. Schokkaert, and J. Flechet, “The incidence of social security contributions: An empirical analysis,” Empirica, vol. 30, no. 2, pp. 81–106, 2003.

[83] L. G. Pee and A. Kankanhalli, “Understanding the drivers, enablers, and performance of knowledge management in public organizations,” in Proc. 2nd Int. Conf. Theory Practice Electron. Govern., pp. 439–466, 2008.

[84] J. F. Quinn and R. V. Burkhauser, “Influencing retirement behaviour: A key issue for social security,” J. Policy Anal. Manag., vol. 3, no. 1, pp. 1–13, 1983.

[85] R. Rofman, “Social security coverage in Latin America,” Social Protection Series Discussion Paper, May 2005.

[86] S. Rosen and P. Taubman, “Changes in life-cycle earnings: What do social security data show?” J. Human Res., vol. 17, no. 3, pp. 321–338, 1982.

[87] N. Rossi and I. Visco, “National saving and social security in Italy,” Ricerche Economiche, vol. 49, pp. 329–356, 1995.

[88] K. Rowlingson and Policy Studies Institute Staff, Social Security Fraud, “The Role of Penalties” London, U.K.: Stationery Office Books, 1997.

[89] B. Rubenstein-Montano, J. Buchwalter, and J. Liebowitz, “Knowledge management: A U.S. social security administration case study,” Govern. Inf. Q., vol. 18, no. 3, pp. 223–253, 2001.

[90] J. Rust and C. Phelan, “How social security and medicare affect retirement behaviour in a world of incomplete markets,” EconWPA, Washington, DC, Working Paper Public Economics 9406005, 1994.

[91] A. A. Samwick, “New evidence on pensions, social security, and the timing of retirement,” National Bureau of Economic Research, Cambridge, MA, Working Paper 6534, 1998.

[92] E. Sheshinski and Y. Weiss, “Uncertainty and optimal social security systems,” Quart. J. Econ., vol. 96, no. 2, pp. 189–206, 1981.

[93] S. Rosen and P. Taubman, “Changes in life-cycle earnings: What do social security data show?” J. Human Res., vol. 17, no. 3, pp. 321–338, 1982.

[94] J. M. Smith, “Viewpoint on public service and computer science,” Commun. ACM, vol. 52, no. 11, pp. 34–35, 2009.

[95] J. E. Stiglitz, “Economics of the Public Sector”, 3rd ed. New York: Norton, 2000.

[96] A. B¨orsch-Supan and R. Schnabel, “Social security and declining laborforce participation inGermany,” Amer. Econ. Rev., vol. 88, no. 2, pp. 173–178, 1998.

[97] P. Orszag and J. E. Stiglitz, Rethinking Pension Reform: Ten Myths About Social Security Systems, World Bank, 1999.

[98] L. H. Thompson, "The social security reform debate," J. Econ. Literature, vol. 21, no. 4, pp. 1425–1467, 1983.

[99] C. Usher, J. Wildfire, and S. Schneider, "Family to family. Tools for rebuilding foster care: The need for self evaluation—Using data to guide policy and practice," Nat. Assoc. College Admission Counseling, Arlington, VA, Rep., 2001.

[100] C. L. Usher, E. Locklin, J. B. Wildfire, and C. C. Harris, "Child welfare performance ratings: One state's approach," Administrat. Soc. Work, vol. 25, pp. 35–51, May 2001. This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination. CAO: SOCIAL SECURITY AND SOCIAL WELFARE DATA MINING: AN OVERVIEW

[101] D. Webster, B. Needell, and J. Wildfire, "Data are your friends: Child welfare agency self-evaluation in Los Angeles county with the family to family initiative," Children Youth Serv. Rev., vol. 24, nos. 6–7, pp. 471–484, 2002.

[102] B. C. Williams, L. B. Demitrack, and B. E. Fries, "The accuracy of the national death index when personal identifiers other than social security number are used.," Amer. J. Public Health, vol. 82, no. 8, pp. 1145–1147, 1992.

[103] S. Wu, Y. Zhao, H. Zhang, C. Zhang, L. Cao, and H. Bohlscheid, "Debt detection in social security by adaptive sequence classification," in Proc. 3rd Int.Conf., Knowl. Sci., Eng.Manage. (ser. Lecture Notes in Computer Science 5914). New York: Springer-Verlag, pp. 192–203, 2009.

[104] P. Zhang, F. Xu, L. Jiang, and R. Ge, "G2c e-government: Shanghai social security and citizen services," in Proc. 7th Int. Conf. Electron. Commerce, pp. 558–563, 2005.

[105] H. Zhang, Y. Zhao, L. Cao, and C. Zhang, "Class association rule mining with multiple imbalanced attributes," in Australian Conf. Artif. Intell. New York: Springer-Verlag, pp. 827–831, (ser. Lecture Notes in Computer Science 4830), 2007.

[106] H. Zhang, Y. Zhao, L. Cao, and C. Zhang, “Combined association rule mining,” in Proc. 12th Pacific-Asia Conf. Adv. Knowl. Discovery Data Mining, pp. 1069–1074, 2008.

[107] H. Zhang, Y. Zhao, L. Cao, and C. Zhang, “Rare class association rule mining with multiple imbalanced attributes,” in Proc. 20th Australian Joint Conf. Adv. Artif. Intell., pp. 827–831, 2009.

[108] H. Zhang, Y. Zhao, L. Cao, C. Zhang, and H. Bohlscheid, “Customer activity sequence classification for debt prevention in social security,” J. Comput. Sci. Technol., vol. 24, no. 6, pp. 1000–1009, 2009.

[109] J. Zhang and J. Zhang, “How does social security affect economic growth? evidence from cross-country data,” J. Populat. Econ., vol. 17, no. 3, pp. 473–500, 2004.

[110] Y. Zhao, H. Zhang, F. Figueiredo, L. Cao, and C. Zhang, “Mining for combined association rules on multiple datasets,” in Proc. Int. Workshop Domain Driven Data Mining, pp. 18–23, 2007.

[111] Y. Zhao, H. Zhang, L. Cao, C. Zhang, and H. Bohlscheid, “Efficient mining of event-oriented negative sequential rules,” in Proc. IEEE/WIC/ACM Int. Conf. Web Intell., pp. 336–342, 2008.

[112] Y. Zhao, H. Zhang, L. Cao, C. Zhang, and H. Bohlscheid, “Combined pattern mining: From learned rules to actionable knowledge,” in Advances in Artificial Intelligence (ser. Lecture Notes in Computer Science 5360). New York: Springer-Verlag, pp. 393–403, 2008.

[113] Y. Zhao, C. Zhang, and L. Cao, “Post-Mining of Association Rules: Techniques for Effective Knowledge Extraction” Hershey, PA: IGI Global, 2009.

[114] Y. Zhao, H. Zhang, L. Cao, H. Bohlscheid, Y. Ou, and C. Zhang, “Data mining applications in social security,” in Data Mining for Business Applications, L. Cao, P. S. Yu, C. Zhang, and H. Zhang, Eds. New York: Springer, pp. 81–96, 2009.

[115] Y. Zhao, H. Zhang, S. Wu, J. Pei, L. Cao, C. Zhang, and H. Bohlscheid, “Debt detection in social security by sequence classification using both positive and negative

patterns," in Machine Learning and Knowledge Discovery in Databases (ser. Lecture Notes Computer Science 5782). New York: Springer-Verlag, pp. 648–663, 2009.

[116] Y. Zhao, H. Zhang, L. Cao, C. Zhang, and H. Bohlscheid, "Mining both positive and negative impact-oriented sequential rules from transactional data, in Advances in Knowledge Discovery and Data Mining" (ser. Lecture Notes Computer Science 5476). New York: Springer-Verlag, pp. 656–663, 2009.

[117] Z. Zheng, Y. Zhao, Z. Zuo, L. Cao, H. Zhang, and C. Zhang, "An efficient GA-based algorithm for mining negative sequential patterns," in Proc. Pacific-Asia Conf. Adv. Knowl. Discovery Data Mining, pp. 262–273, 2010.

[118] Z. Zheng, Y. Zhao, Z. Zuo, and L. Cao. "Negative-GSP: An efficient method for mining negative sequential patterns", in Proc. AusDM pp. 63–67, 2009.

[119] Australian National Audit Office, ANAO Centrelink Audit Report 2007–08. Canberra, Australia: Commonwealth, 2008.

[120] Centrelink, Integrated Activity Management Developer Guide, Canberra, Australia: Commonwealth, 1999.

[121] Centrelink, Annual Report 2009, Canberra, Australia: Commonwealth, 2009.

[122] Department of Human Services, "Better Dealings with Government: Innovation in Payments and Information Services", Canberra, Australia: Commonwealth, 2009.

[123] "The data measures, data composites, and national standards to be used in child and family services reviews", Attachment B: "Methodology for developing the composites," U.S. Department of Health and Human Services, Washington, DC, 2006.

[124] "Administration for children & families," child Welfare Monitoring, [Online]. Available: http://www.acf.hhs.gov/programs/cb/cwmonitoring/ index.htm, 2007.

[125] " National Resource Centre for Child Welfare Data and Technology" [Online]. Available: http://www.nrccwdt.org/index.html, 2007.

[126] “Centrelink Customer Risk Rating at the Initial Registration”, UTS Centrelink Contract Research Project, 2009.

[127] “The Provision of Income Reporting Data Analysis Services”, UTS Centrelink Contract Research Project, 2006.

[128] “Pattern Analysis and Risk Control of E-Commerce Transactions to Secure Online Payments”, Australian Research Council Linkage Grant, 2007–2009.

[129] “Centrelink Fraud Investigation: Opportunities and Test”, UTS-Centrelink Contract Research Project, 2010.

[130] ARC Linkage, “Data Mining of Activity Transactions to Strengthen Debt Prevention”, Australian Research Council Linkage Grant, 2007–2009.

[131] ARC Linkage, “Detecting Significant Changes in Organisation Customer Interactions Leading to Non-Compliance”, Australian Research Council Linkage Grant, 2010–2013.

[132] “Detecting Incorrect Income Declaration in Real Time”, UTS-Centrelink Contract Research Project, 2010.

[133] [Online]. Available: http://www.centrelink.gov.au/internet/internet.nsf/about_us /fraud_index. Htm, 2009.

[134] [Online]. Available: http://www.skipease.com/blog/datamining/data-mining-detects-welfare-fraud, 2009.

[135] R. Agrawal and R. Srikant, “Fast algorithms for mining association rules in large databases” in J. B. Bocca, M. Jarke, and C. Zaniolo (Editors). Proceedings of the 20th International Conference on Very Large Data Bases, VLDB, pp. 487–499, Santiago, Chile, 1994.

[136] J. Pei, J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, U. Dayal, and M.C. Hsu, “Mining sequential patterns by pattern-growth: The prefixspan approach,” IEEE Transactions on Knowledge and Data Engineering, 16(11): pp. 1424–1440, 2004.

[137] Q. Yang, J. Yin, C. X. Ling, and T. Chen, “Postprocessing decision trees to extract actionable knowledge,” in ICDM 2003: Proceedings of the Third IEEE International Conference on Data Mining, page 685, Washington, DC, USA, IEEE Computer Society, 2003.

[138] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in KDD, pp. 226–231, 1996.

[139] A. Silberschatz and A. Tuzhilin, “What makes patterns interesting in knowledge discovery systems,” IEEE Transactions on Knowledge and Data Engineering, 8(6): pp.970–974, 1996.

[140] M. Zaki, “Mining non-redundant association rules’” Data Mining and Knowledge Discovery, 9: pp.223–248, 2004.

[141] H. Guo and H. L. Viktor, “Learning from imbalanced data sets with boosting and data generation: the databoost-im approach” SIGKDD Explor. Newsl., 6(1): pp.30–39, 2004.

[142] J. Zhang, E. Bloedorn, L. Rosen, and D. Venese, “Learning rules from highly unbalanced data sets,” in ICDM 2004: Proceedings of the Fourth IEEE International Conference on Data Mining, pp. 571–574, Washington, DC, USA, IEEE Computer Society, 2004.

[143] J. Chattratichat, J. Darlington, Y. Guo, S. Hedvall, M. K?ler, and J. Syed, “An architecture for distributed enterprise data mining,” in HPCN Europe 2099: Proceedings of the 7th International Conference on High-Performance Computing and Networking, pp. 573–582, London, UK, 1999.

[144] V. Crestana-Jensen and N. Soparkar, “Frequent itemset counting across multiple tables” in PAKDD 2000: Proceedings of the 4th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Current Issues and New Applications, pp. 49–61, London, UK, Springer-Verlag, 2000.

[145] L. Cristofor and D. Simovici, “Mining association rules in entity-relationship modelled databases”, Technical report, University of Massachusetts Boston, 2001.

[146] P. Domingos, “Prospects and challenges for multi-relational data mining”, SIGKDD Explor. Newsl., 5(1): pp. 80–83, 2003.

[147] G. Dong and J. Li, “Efficient mining of emerging patterns: discovering trends and differences”’ in KDD 1999: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 43–52, New York, NY, USA, 1999.

[148] S. Dzeroski, “Multi-relational data mining: an introduction”, SIGKDD Explor. Newsl., 5(1): pp.1–16, 2003.

[149] B. Park and H. Kargupta, “Distributed data mining: Algorithms, systems, and applications”’ in N. Ye, editor, Data Mining Handbook. 2002.

[150] F. Provost, “Distributed data mining: Scaling up and beyond”’ in Advances in Distributed and Parallel Knowledge Discovery, MIT Press, 2000.

[151] R. Agrawal, T. Imielinski, and A. Swami, “Mining association rules between sets of items in large databases”, in Proc. of the ACM SIGMOD International Conference on Management of Data, pp. 207–216, Washington D.C. USA, 1993.

[152] R. Agrawal and R. Srikant, “Mining sequential patterns”, in P. S. Yu and A. S. P. Chen, editors, Proc. of the 11th International Conference on Data Engineering, pp. 3–14, Taipei, Taiwan, IEEE Computer Society Press, 1995.

[153] A. Savasere, E. Omiecinski, and S.B. Navathe, “Mining for strong negative associations in a large database of customer transactions”, in ICDE 1998: Proc. of the 14th International Conference on Data Engineering, pp. 494–502, Washington, DC, USA, 1998. IEEE Computer Society, 1998.

[154] X. Yuan, B. P. Buckles, Z. Yuan, and J. Zhang, “Mining negative association rules”, in ISCC ’02: Proc. of the 17th International Symposium on Computers and Communications (ISCC’02), page 623, Washington, DC, USA, 2002. IEEE Computer Society.

[155] M.L. Antonie and O.R. Zane, “Mining positive and negative association rules: an approach for confined rules”, in PKDD ’04: Proc. of the 8th European Conference on

Principles and Practice of Knowledge Discovery in Databases, pp. 27–38, New York, NY, USA, 2004. Springer-Verlag New York, Inc.

[156] X. Wu, C. Zhang, and S. Zhang, “Efficient mining of both positive and negative association rules”, in ACM Transactions on Information Systems, Vol. 22(No.3): pp.381–405, 2004.

[157] H. Bannai, H. Hyyro, A. Shinohara, M. Takeda, K. Nakai, and S. Miyano’ “Finding optimal pairs of patterns”, in Proc. of the 4th Workshop on Algorithms in Bioinformatics (WABI’04), 2004.

[158] N.P. Lin, H.J. Chen, and W.H. Hao, “Mining negative sequential patterns”’ in: Proc. of the 6th WSEAS International Conference on Applied Computer Science, pp. 654–658, Hangzhou, China, 2007.

[159] W.M. Ouyang and Q.H. Huang, “Mining negative sequential patterns in transaction databases”’ in Proc. of 2007 International Conference on Machine Learning and Cybernetics, pp. 830–834, Hong Kong, China, 2007.

[160] X. Sun, M. E. Orlowska, and X. Li, “Finding negative event-oriented patterns in long temporal sequences”, in H. Dai, R. Srikant, and C. Zhang, editors, PAKDD, volume 3056 of Lecture Notes in Computer Science, pp. 212–221, Springer, 2004.

[161] R. Srikant and R. Agrawal, “Mining sequential patterns: Generalizations and performance improvements”, in EDBT 2096: Proc. of the 5th International Conference on Extending Database Technology, pp. 3–17, London, UK, 1996. Springer-Verlag.

[162] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M.C. Hsu, “Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth” in ICDE 2001, Proc. of the 17th International Conference on Data Engineering, page 215,Washington, DC, USA, IEEE Computer Society, 2001

[163] M.J. Zaki, “Spade: An efficient algorithm for mining frequent sequences. Machine Learning”, 42(1-2): pp.31–60, 2001.

[164] J. Ayres, J. Flannick, J. Gehrke, and T. Yiu, “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, New York, NY, USA, 2002. ACM, 2002.

[165] J. Han, J. Pei, B. Mortazavi-Asl, Q. Chen, U. Dayal, and M.C. Hsu, “Freespan: frequent pattern-projected sequential pattern mining”, in KDD 2000: Proc. Of the 6th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 355–359, New York, NY, USA, 2000. ACM, 2000.

[166] F. Verhein and S. Chawl, “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.

[167] W. Li, J. Han, J. Pei and 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.

[168] B. Liu, W. Hsu and Y. Ma “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.

[169] P.N. Tan, V. Kumar and J. Srivastava, “Selecting the right interestingness measure for association patterns”, in Proc. the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD2002), Edmonton, Canada, July 23–26, pp. 32–41, 2002.

[170] B. Lent, A.N. Swami and J. Widom, “Clustering association rules”, in Proceedings of the 13th International Conference on Data Engineering, Birmingham, U.K, April 7–11, pp. 220–231. IEEE Computer Society, Los Alamitos, 1997.

[171] B. Liu, W. Hsu and Y. Ma, “Pruning and summarizing the discovered associations”, in Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 1999), pp. 125–134. ACM Press, New York,1999.

[172] R.J. Hilderman, C. L. Carter, H. J. Hamilton and N. Cercone, “Mining Market Basket Data Using Share Measures and Characterized Itemsets”, in X. Wu, R. Kotagiri, K. B. Korb (Eds.) PAKDD 1998. LNCS, vol. 1394, pp. 159–170, Springer, Heidelberg, 1998.

[173] B. Liu and W. Hsu, “Post-analysis of learned rules”, in Proceedings of the 13th National Conference on Artificial Intelligence (AAAI 1996), Portland, Oregon, USA, pp. 828–834, 1996.

[174] O. R. Zane and M. L.Antonie, “On pruning and tuning rules for associative classifiers”, in R. Khosla, R. J. Howlett and L. C. Jain (Eds.) KES 2005. LNCS (LNAI), vol. 3683, pp. 966–973. Springer, Heidelberg, 2005.

[175] Y. Zhao, L. Cao, Y. Morrow, Y. Ou, J. Ni and C. Zhang, “Discovering debtor patterns of Centrelink customers”, in Proc. of The Australasian Data Mining Conference: AusDM 2006, Sydney, Australia, 2006.

[176] “Centrelink Fraud Statistics and Centrelink Facts and Figures”, url: http://www.centrelink.gov.au/internet/internet.nsf/about us/fraud stats.htm, http://www. centrelink.gov.au/internet/internet.nsf/about us/facts.htm, 2006.