软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2020,31(3):710−725 [doi: 10.13328/j.cnki.jos.005905]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

面向多维稀疏数据仓库的欺诈销售行为挖掘*

郑皎凌[1,2], 乔少杰[1,2], 舒红平[1,2], 应广华[3], Louis Alberto GUTIERREZ[4]

[1](软件自动生成与智能服务四川省重点实验室(成都信息工程大学),四川 成都 610225)

[2](成都信息工程大学 软件工程学院,四川 成都 610225)

[3](阿里巴巴技术有限公司,浙江 杭州 311121)

[4](Department of Computer Science, Rensselaer Polytechnic Institute, New York, USA)

通讯作者: 乔少杰, E-mail: sjqiao@cuit.edu.cn

摘 要: 分销渠道系统中,产品制造商会分配给销售额较大的分销商更多返点利润鼓励销售,而分销商之间可能会联合起来将多个分销商的销售业绩累计在其中一个分销商上,获取高额利润,这种商业欺诈行为被称为挂单或窜货.由于数据中大量正常极值点的存在,使得传统异常探测算法很难区分正常极值和由挂单导致的异常极值;另外,多维销售数据本身就存在的稀疏性导致多维数据异常探测算法无法有效运行.为了克服上述问题,将人工智能和数据库技术结合起来,提出了基于分割率的特征提取方法和基于张量重构的挂单行为挖掘算法.同时,由于分销商之间存在多种挂单行为,设计了基于挂单模式偏序格的特征提取方法来对销售数据集中存在的挂单行为进行分类.在合成数据的实验中,所提出的挂单点挖掘算法能达到 65%的平均 AUC 值,而传统特征提取方法仅达到 36%和 30%的平均 AUC 值.在真实数据上的实验结果表明,挂单行为探测方法能区分正常销售极值和挂单行为产生的异常极值.

关键词: 分析渠道欺诈;人工智能;挂单模式;张量;偏序格

中图法分类号: TP18

中文引用格式: 郑皎凌,乔少杰,舒红平,应广华,Gutierrez LA.面向多维稀疏数据仓库的欺诈销售行为挖掘.软件学报,2020,31(3):710−725. http://www.jos.org.cn/1000-9825/5905.htm

英文引用格式: Zheng JL, Qiao SJ, Shu HP, Ying GH, Gutierrez LA. Sale fraud behavior detection over multidimensional sparse data warehouse. Ruan Jian Xue Bao/Journal of Software, 2020,31(3):710−725 (in Chinese). http://www.jos.org.cn/1000-9825/5905.htm

Sale Fraud Behavior Detection over Multidimensional Sparse Data Warehouse

ZHENG Jiao-Ling[1,2], QIAO Shao-Jie[1,2], SHU Hong-Ping[1,2], YING Guang-Hua[3], Louis Alberto GUTIERREZ[4]

[1](Sichuan Key Laboratory of Software Automatic Generation and Intelligent Service (Chengdu University of Information Technology), Chengdu 610225, China)

* 基金项目: 国家自然科学基金(61772091, 61802035, 61962006); 四川省科技计划(20YYJC2785, 2018JY0448, 2019YFG0106, 2019YFS0067); 四川高校科研创新团队建设计划(18TD0027); 广西自然科学基金(2018GXNSFDA138005); 成都信息工程大学科研基金(KYTZ201715, KYTZ201750); 成都信息工程大学中青年学术带头人科研基金(J201701); 广东省普及型高性能计算机重点实验室项目(2017B 030314073)

Foundation item: National Natural Science Foundation of China (61772091, 61802035, 61962006); Sichuan Science and Technology Program (20YYJC2785, 2018JY0448, 2019YFG0106, 2019YFS0067); Innovative Research Team Construction Plan in Universities of Sichuan Province (18TD0027); National Natural Science Foundation of Guangxi of China (2018GXNSFDA138005); Scientific Research Foundation for Advanced Talents of Chengdu University of Information Technology (KYTZ201715, KYTZ201750); Scientific Research Foundation for Young Academic Leaders of Chengdu University of Information Technology (J201701); Guangdong Province Key Laboratory of Popular High Performance Computers (2017B030314073)

本文由人工智能赋能的数据管理、分析与系统专刊特约编辑李战怀教授、于戈教授和杨晓春教授推荐.

收稿时间: 2019-07-20; 修改时间: 2019-09-10; 采用时间: 2019-11-25; jos 在线出版时间: 2020-01-10

CNKI 网络优先出版: 2020-01-10 13:34:41, http://kns.cnki.net/kcms/detail/11.2560.TP.20200110.1334.006.html

[2](School of Software Engineering, Chengdu University of Information Technology, Chengdu 610225, China)
[3](Alibaba (China) Technology Co. Ltd., Hangzhou 311121, China)
[4](Department of Computer Science, Rensselaer Polytechnic Institute, New York, USA)

Abstract: In distribution channel system, product manufacturer will often reward retail trader who makes big deal to increase the sales. On the other hand, in order to obtain high reward, retail traders may form alliance, where a cheating retail trader accumulates the deals of other retail traders. This type of commercial fraud is called deal cheating or cross region sale. Because the sales contain a lot of normal big deals, traditional outlier detection methods cannot distinguish the normal extreme value and the true outlier generated by deal cheating behavior. Meanwhile, the sparsity of the multidimensional sales data makes the outlier detection methods based on multidimensional space cannot work effectively. To handle the aforementioned problems, this study proposes deal cheating mining algorithms based on ratio characteristic and tensor reconstruction method. These algorithms combine artificial intelligence and database technique. Meanwhile, because there are multiple types of deal cheating patterns, this study proposes deal cheating pattern classification methods based on the partially ordered lattice of deal cheating patterns. In the experiments on synthetic data, the deal cheating detection algorithm based on the ratio characteristic can achieve an average AUC-value of 65%. The traditional feature extraction methods can only achieve average AUC-values of 36% and 30%. In the experiments on the real data, the results shows the deal cheating detection algorithm is capable of distinguishing normal big deal from abnormal big deal which may be generated by the deal cheating behaviors.

Key words: distribution channel fraud; artificial intelligence; deal cheating pattern; tensor; partially ordered lattice

在移动互联网时代,随着越来越多的交易和操作转移到了线上,也出现了各种各样的欺诈行为.电商行业中出现了刷单现象,公众服务行业出现了黄牛倒卖现象,O2O 行业出现了垃圾小号现象等,这些欺诈行为已经形成了所谓的“黑色产业”,从业者通过线上线下的商业漏洞来牟利.相似的商业欺诈行为也发生在分销渠道系统中.在该系统中,大品牌公司不会将其产品直接卖给消费者,而是选择一些分销商将其产品销售给最终客户,并且制定了一系列激励措施激励那些产生大额销量的分销商.这导致了欺诈行为的产生,多个分销商会联合起来将自己的销售额累计在其中一个分销商身上,这种分销渠道上的欺诈行为被称为挂单.

挂单分析的典型应用场景是线上家电产业窜货分析.随着电商产业不断发展,窜货行为在线上交易中日益盛行,并开始对线下产业造成危害,这种危害在家电行业中体现得尤为突出.因为在家电行业中同一产品在分销渠道不同的地区销售价格是不同的.但在电子商务平台上,小经销商会在不同的地区以相同的价格销售相同的产品.当线上销量增大时,小型分销商会积累不同地区的产品,并在需求量大的地区进行销售.这种销售积累行为将使小分销商获得更多的利润,但却违反了家电行业分销渠道的销售规则,可以说是一种典型的分销渠道欺诈行为,即俗称的窜货.这种行为在电商平台上发展壮大,将会对市场带来很大的负面影响.

虽然在 1989 年就有研究[1]详细描述过分销渠道系统挂单的概念和方式,但电商使挂单行为更隐蔽和更容易实施.本文提出的挖掘算法旨在帮助审计部门在大数据场景下快速检测挂单欺诈行为,是结合日益增多的线上电商销售业务真实应用提出的,是一项非常困难和富有挑战意义的新课题,主要存在以下两个难点.

(1) 销售数据仓库的数据方体存在稀疏性.由于分销商不可能在每个时间点对每种商品都有销售,所以产生了数据稀疏的问题.当数据仓库中数据方体的维度是较宏观的概念级别,此时不存在为空的数据方体;而当数据仓库中数据方体的维度是较微观的概念级别,就会存在大量空的数据方体,无法采用基于数据仓库的联机分析处理技术(online analytical processing,简称 OLAP);

(2) 正常极值和异常极值的同时存在.由于商品的销售额本来就服从幂律分布,即 80%的销售额由 20%的商品产生的,如促销、明星产品、节假日(双十一等)等,都会导致某些商品极高的销售额,这些极值是正常商业行为产生的.所以,正常极值和异常极值的混合,将使得传统基于极值异常检测的方法很难有效工作.

为弥补现有方法的不足,本文提供了针对多维稀疏销售数据仓库的欺诈销售行为挖掘方法,主要贡献有:

(1) 提出了多维数据仓库中数据块的概念,通过数据块的维度变化来定义不同的挂单模式和挂单点;

(2) 提出了数据块的度量指标,称为分割率,它不会受到数据方体中数据稀疏性的影响.基于这一概念,我们可以将数值异常检测和多维数据空间异常检测方法相结合,来发现存在挂单行为的销售记录;

(3) 提出了挂单模式偏序格的概念,通过引入偏序格中各个挂单模式的相对位置偏序结构信息,有效地使用了数据仓库中的维度层次信息来对挂单行为所遵循的挂单模式进行挖掘;

(4) 在真实销售数据上进行了大量实验,验证了算法的准确率、时间效率等指标.

为了清楚地解释挂单问题本身的新颖性,本文详细描述了整个挂单过程和检测过程,同时在相关工作中具体分析了每一步和现有类似欺诈检测问题的异同.

为了清楚地解释数据块和分割率在问题中的工作原理和重要性,本文构造了具体的欺诈挂单行为及具体的欺诈检测过程,通过与其他几种方法的对比,证明了数据块与分割率的优势.

本文第 1 节对本文要求解的问题进行形式化描述.第 2 节阐述该领域的相关工作.第 3 节阐述对问题 1 的求解方法.第 4 节阐述对问题 2 的求解方法.第 5 节给出在模拟数据和真实数据上的实验结果.第 6 节总结本文的工作.

1 问题描述

1.1 问题的定义

分销商渠道欺诈行为挖掘包含两层含义:一是挂单模式挖掘,因为分销商之间的挂单行为不是随机的,而是遵循某种规则,比如挂单行为只能在相同类型的商品之间进行或相同类型的分销商之间进行等;二是挂单点挖掘,当确定了挂单模式之后,要挖掘出那些作为销售额累积点的销售记录,称为挂单点.为了准确描述本文所要求解的问题,先给出以下定义.

定义 1(多维数据空间 $\boldsymbol{D}=(\boldsymbol{A_1},\boldsymbol{A_2},\ldots,\boldsymbol{A_n})$). D 由 n 个维度构成,$A_i=\{a_{i1},a_{i2},\ldots,a_{im}\}$,$A_i$ 中的每个元素代表第 i 维上的一个概念级别.

定义 2(多维数据空间 $\boldsymbol{D}$ 上的偏序格 $\boldsymbol{L}=\langle \boldsymbol{M},\preccurlyeq\rangle$). 设 $D=(A_1,A_2,\ldots,A_n)$,$M=\{l_1,l_2,...,l_m\}$,对任意 $l\in M$,$l=(a_1,a_2,\ldots,a_n)$,其中,$a_i\in A_i$,称 l 为偏序格 L 的格点.对于 M 中的任意两个格点 $l_i=(a_{i1},a_{i2},\ldots,a_{in})$和 $l_j=(a_{j1},a_{j2},\ldots,a_{jn})$,如果 $l_i\preccurlyeq l_j$,表示 l_i 在各维度上的级别均低于或等于 l_j 在相应维上的概念级别.

定义 3(销售数据仓库 $\boldsymbol{R}$). $R=\{t_1,...,t_N\}$是包含 N 条销售记录的销售数据集,设 R 所在的多维空间 $D=\{A_1,A_2,\ldots,A_n\}$,对任意 $t_i\in R$,有 $t_i=v(\alpha_1,\alpha_2,...,\alpha_n,s)$,其中,$v$ 是 t_i 的销售额,$(\alpha_1,\alpha_2,...,\alpha_n)$是 t_i 的记录属性在 D 上各个维度和概念层次的取值,s 是产生该销售记录的分销商 ID.

定义 4(销售数据仓库 $\boldsymbol{R}$ 在 $\boldsymbol{D}$ 上的数据分块 $\boldsymbol{Chunk}(\boldsymbol{R},\boldsymbol{l})$). 设 $R=\{t_1,...,t_N\}$,L 是 D 上的偏序格 $L=\langle M,\preccurlyeq\rangle$,$l\in M$ 并且 $l=(a_1,a_2,\ldots,a_n)$,则 $Chunk(R,l)=\{C_1,C_2,...,C_k\}$,$C_i\cap C_j=\varnothing$,$C_1\cup C_2\cup...\cup C_k=R$,对任意 $t_i=v(\alpha_{i1},\alpha_{i2},...,\alpha_{in},s_i)$,$t_j=v(\alpha_{j1},\alpha_{j2},...,\alpha_{jn},s_j)$,如果 $t_i\in C_i$ 并且 $t_j\in C_i$,则$(\alpha_{i1},\alpha_{i2},...,\alpha_{in})=(\alpha_{j1},\alpha_{j2},...,\alpha_{jn})$;否则,$(\alpha_{i1},\alpha_{i2},...,\alpha_{in})\neq(\alpha_{j1},\alpha_{j2},...,\alpha_{jn})$.

定义 5(挂单行为 $\boldsymbol{g}(\boldsymbol{t_1},\boldsymbol{t_2})$). 设有两条销售记录数据 $t_i=v(\alpha_{i1},\alpha_{i2},...,\alpha_{in},s_i)$,$t_j=v(\alpha_{j1},\alpha_{j2},...,\alpha_{jn},s_j)$,如果 t_1,t_2 之间存在挂单行为,则在挂单行为发生后,$t_i=v'(\alpha_{i1},\alpha_{i2},...,\alpha_{in},s_i)$,$t_j=v'(\alpha_{j1},\alpha_{j2},...,\alpha_{jn},s_j)$,并且有 $v'(\alpha_{i1},\alpha_{i2},...,\alpha_{in},s_i)>>v(\alpha_{i1},\alpha_{i2},...,\alpha_{in},s_i)$以及 $v'(\alpha_{j1},\alpha_{j2},...,\alpha_{jn},s_j)<<v(\alpha_{j1},\alpha_{j2},...,\alpha_{jn},s_j)$,称 t_i 为被挂单记录,t_j 为挂单记录.

定义 6(挂单模式). 设有销售数据仓库 R,R 所在的多维空间 $D=\{A_1,A_2,\ldots,A_n\}$,$L=\langle M,\preccurlyeq\rangle$是 D 上的偏序格,$M=\{l_1,l_2,\ldots,l_m\}$,称 $l(l\in M)$为 R 上的一个挂单模式,称 $g(t_1,t_2)$为挂单模式 l 下的挂单行为当且仅当 $t_1\in C_i,t_2\in C_i$,其中,$C_i\in Chunk(R,l)$.

图 1(a)画出了多维空间 D=(分销商,商品,时间)的概念层次以及相应的偏序格 L,由于篇幅原因省略了时间维度.图 1(b)是 D 上的销售数据仓库 $R=\{t_1,...,t_{18}\}$,设每个维层次结构分别为分销商 ID→分销商类型→All,商品 ID→商品系列→商品品牌→商品类型→All;月份→All,则 D 上的偏序格共有 3×5×2=30 个格点为描述简洁.可知 l={商品类型,分销商类型,*}是 L 的一个格点,则 $Chunk(R,l)=\{C_1,C_2,C_3,C_4\}$,$C_1$=(美妆店,飘柔,*)=$\{t_1,t_2,t_3,t_4,t_5,t_6\}$,$C_2$=(美妆店,潘婷,*)=$\{t_7,t_8,t_9,t_{10}\}$,$C_3$=(批发市场,飘柔,*)=$\{t_{11},t_{12},t_{13},t_{17}\}$,$C_4$=(批发市场,潘婷,*)=$\{t_{14},t_{15},t_{16},t_{18}\}$.图中箭头表示一次挂单行为,分别是 $g(t_1,t_3)\in C_1,g(t_1,t_5)\in C_1,g(t_2,t_4)\in C_1$.后文图 2(a)中给出了每个销售记录的具体金

额,单位为万元,t_1,t_2,t_3,t_4,t_5 为挂单记录,黄色单元格详细展示了挂单前后 t_1,t_2,t_3,t_4,t_5 的数值变化.

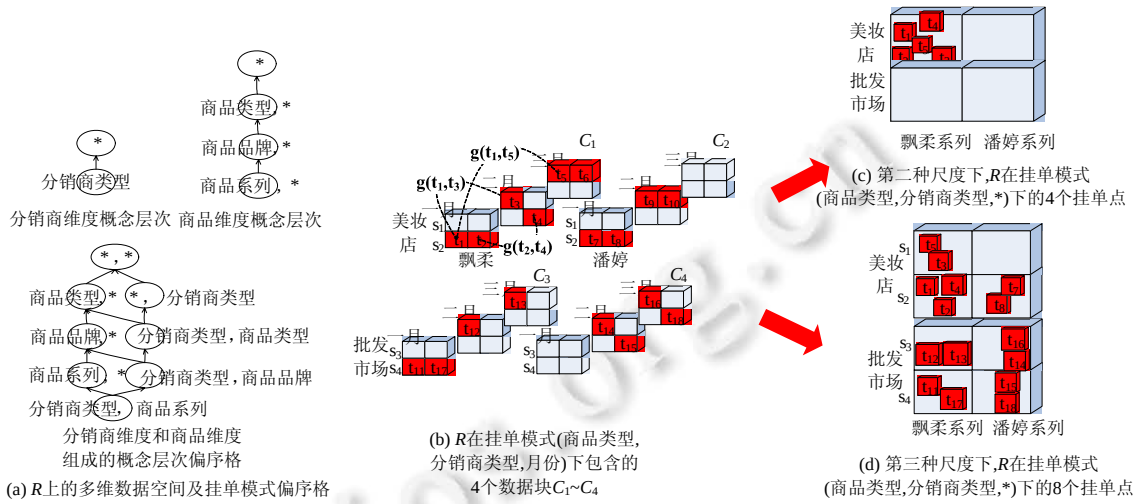


Fig.1 Sale accumulation pattern example

图 1 挂单模式示例

根据定义 1~定义 6 可以给出本文待求解问题的定义.

问题 1(分销商挂单模式挖掘). 设有销售数据仓库 $R,L=\langle M,\preccurlyeq\rangle$是多维数据空间 D 上的偏序格,$M=\{l_1,...,l_n\}$是 R 上所有可能的挂单模式集合,设已知 R 中存在的挂单行为是 $l(l\in M)$,分销商挂单模式挖掘旨在找出从 M 中找出 R 上的真实挂单模式 l.

当知道了 R 的挂单模式,就可以挖掘 R 中所包含的挂单记录.但由于数据的稀疏性和正常销售极值的存在,使得挖掘具体的被挂单记录非常困难,故我们将挖掘挂单记录这个问题转换成挖掘挂单点的问题.

在本问题中,挂单点 p 可以有 3 种尺度:第 1 种是 $p=t$,即挂单点就是被挂单记录;第 2 种是 $p=C_i$,即挂单点是数据块 C_i,并且 C_i 中包含被挂单记录.第 1 种尺度最精确,但较高的挖掘难度会导致很低的挖掘精度;第 2 种尺度的挖掘难度远小于第 1 种,但由于 C_i 中通常会包含很多记录,使得很难从 C_i 中找出真正的被挂单记录.因此,本文设计了第 3 种尺度 $p=C_{i,s}$,$C_{i,s}$ 是 C_i 中的一个子集,只包含 C_i 中属于分销商 s 的销售记录集合.第 3 种尺度下的挂单点规模在第 1 种、第 2 种尺度之间,可以较好地平衡精度和召回率.

定义 7(挂单点). 设有销售数据仓库 R,R 所在的多维空间 $D=\{A_1,A_2,...,A_n\}$,$L=\langle M,\preccurlyeq\rangle$是 D 上的偏序格,$M=\{l_1,l_2,...,l_m\}$.设 R 的挂单模式为 l,则 R 在 l 下的任意挂单点记为 $p_{C_{i,s}}$,$p_{C_{i,s}}=\{t_1,...,t_k\}$,$C_i\in Chunk(l)$.对任意 $t_i=v(\alpha_{i1},\alpha_{i2},...,\alpha_{in},s_i)\in p_{C_{i,s}}$,都有 $t_i\in C_i$ 并且 $s_i=s$.

问题 2(特定挂单模式 l 下的挂单点 p 挖掘). 设已知销售数据仓库 R 上的挂单模为 l,$Chunk(R,l)=\{C_1,C_2,...,C_n\}$,该问题要找出挂单模式 l 下前 k 个最有可能的挂单点.

例 1(不同尺度下的挂单点):在第 1 种尺度下共 18 个挂单点,每个销售记录分别是一个挂单点,其中,t_1,t_2 是真实挂单点,如图 1(b)所示.在第 2 种尺度下共 4 个挂单点,分别是$\{t_1,t_2,t_3,t_4,t_5,t_6\}$,$\{t_7,t_8,t_9,t_{10}\}$,$\{t_{11},t_{12},t_{13},t_{17}\}$,$\{t_{14},t_{15},t_{16},t_{18}\}$,其中,$\{t_1,t_2,t_3,t_4,t_5,t_6\}$是真实挂单点,如图 1(c)所示.在第 3 种尺度下共 8 个挂单点,分别是$\{t_3,t_5,t_6\}$,$\{t_1,t_2,t_4\}$,$\{t_9,t_{10}\}$,$\{t_7,t_8\}$,$\{t_{12},t_{13}\}$,$\{t_{11},t_{17}\}$,$\{t_{14},t_{16}\}$,$\{t_{15},t_{18}\}$,其中,$\{t_1,t_2,t_4\}$是真实挂单点,如图 1(d)所示.

图 1(a)描述了数据仓库 R 中的稀疏性问题.每个挂单模式下有两个数,右边的数字表示该挂单模式包含的总数据块个数,左边的数字表示非空数据块个数.可见当挂单模式位于偏序格的上方时,即维度属性处于概念层次中较宏观的级别,此时挂单模式包含的数据块较少,不存在空数据块;而当挂单模式位于偏序格下方,即维度属性处于概念层次中较微观的级别,挂单模式包含大量空数据块.

1.2 问题求解框架

一般来说,欺诈行为挖掘需要一系列模块共同协作,本文所提出基于数据块、分割率等技术的欺诈挂单行为挖掘也不例外.为了更好地进行挖掘,本文提出一种新的欺诈挂单挖掘框架.图 2(a)是获得含有欺诈挂单过程的多维稀疏销售数据集;图 2(b)根据不同的挂单模式将数据集分割成不同的数据块;图 2(c)基于数据块构造挂单点;图 2(d)~图 2(h)计算不同挂单模式下各个挂单点的异常度;图 2(i)基于每种挂单模式上的挂单点异常度提取挂单模式分类特征;最后构造挂单模式分类模型.

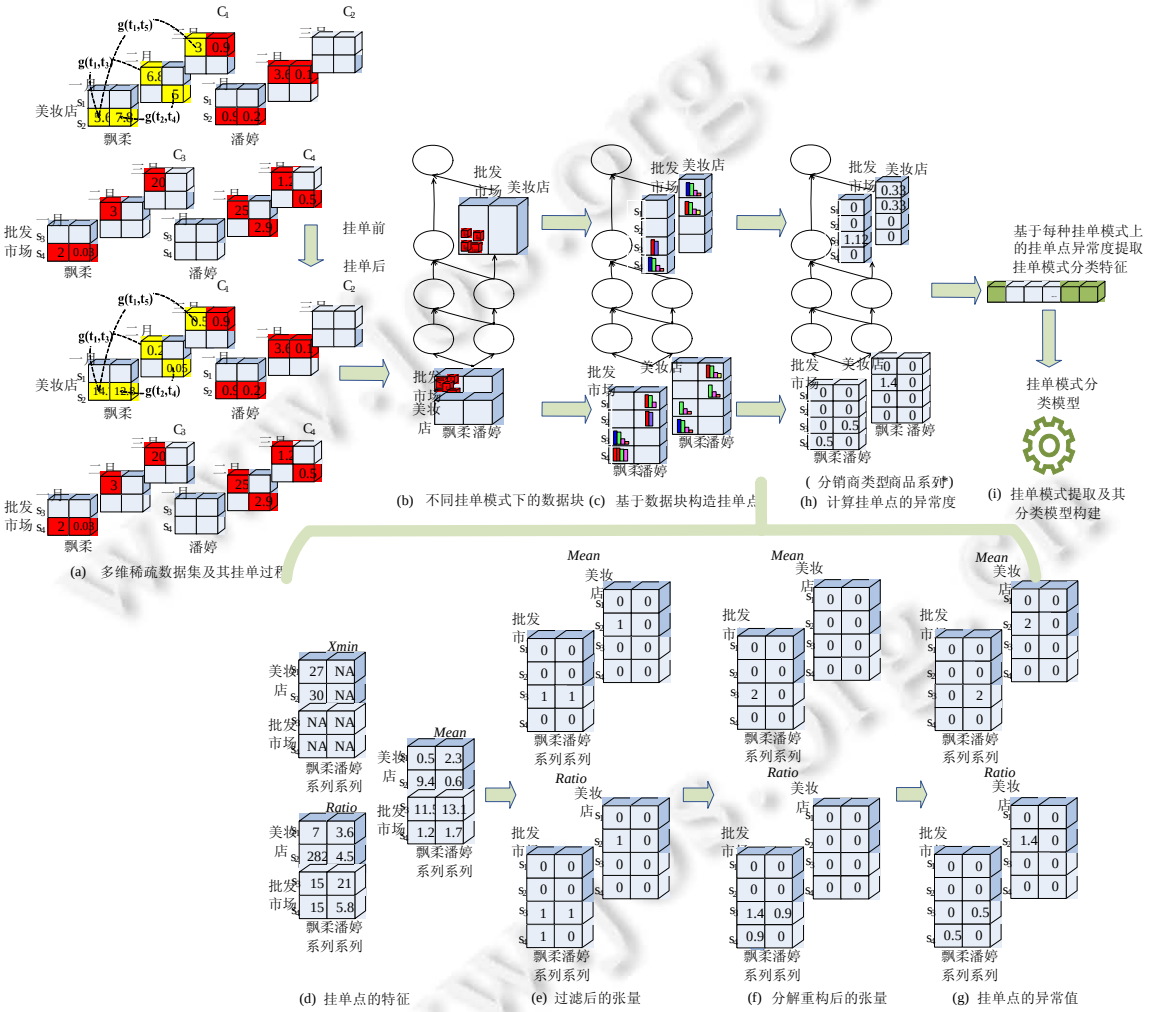


Fig.2 Solution framework based on example

图 2 基于具体实例的总体求解框架

该求解框架的可以将数据仓库多维分析和人工智能技术结合起来,通过统计不同空间维度各数据块所包含销售数据的分布,提取人工智能算法的特征属性,克服了数据稀疏性和正常极值点造成的影响,这是本文在人工智能和数据库两种技术结合过程中做出的探索.

2 相关工作

从所要解决的问题来看,本文属于欺诈检测的范畴.欺诈检测是一项非常有意义并且实用性很强的工作,能

够广泛地应用于银行、保险、政府机构和执法部门等.近年来,在电子商务中的欺诈行为尤为突出,并且大幅增加,这使得欺诈检测比以往任何时候都更加重要.尽管相关机构做出了努力,但每年仍有数亿美元因欺诈而损失.在保险方面,欺诈可以是夸大的损失,或者故意造成支付事故等.近年来,有 25%的索赔包含某种形式的欺诈,导致大约 10%的保险赔付金属于欺诈赔付.因此,快速检测作弊行为以尽可能减少客户的损失是非常重要的.

在已有的欺诈检测研究中,Shu 等人的研究[2]是与本文最相似的工作,也是进行分销商挂单行为分析,但其具体的求解方式和文本是完全不同的:首先,其算法要求数据不能存在稀疏性;其次,算法要求分销商之间只存在一种挂单模式;同时,其所采用的具体检测方法和本文也十分不同.所以本文和该工作只在图 2(a)的欺诈形成过程部分是相似的,在其他部分都是不同的.当然,随着欺诈问题的不同,研究人员设计了一系列不同的检测方法.比如:由于在线用户具有一些固定的移动设备使用习惯,如跨屏行为、聊天、视频观看和点击行为等,Zhang 等人[3]提出了一种序贯行为数据的特征提取框架来检测在线欺诈;Roux 等人[4]提出了一种无监督学习的方法来检测税收问题中的低报上税金额欺诈,节省了基于有监督学习的样本采集工作;此外,由于交互式问答也可能包含用于识别用户信用风险的重要信息,Song 等人[5]提出了基于交互式问答的欺诈特征提取框架,检测在线借贷风险.另一方面,传统风险控制中使用风险评分模型旨在模拟个人的特征,但很难实现对群体风险的全面控制,如帮派欺诈、群体攻击等.为了检测群体欺诈行为,Min 等人[6]提出了一种基于图模型进行特征提取的行为语言处理模型,并将该模型用于检测群体性的在线借贷欺诈行为.在群体风险控制领域,欺诈检测主要集中在发现公司、代理商甚至软件的异常行为.Vlasselaer 等人[7]提出了一种检测公司欺诈性破产以进行逃税的方法.Vlasselaer 等人[8]在检测公司欺诈性破产的过程中,发现欺诈性公司通常隶属于某个欺诈集团.如有隶属于欺诈集团的 3 个即将破产公司 *A*,*B* 和 *C*,他们同时将资源现在转移到集团另一个活跃公司 *D*,而 *D* 在获取资源后在未来短期也进行了欺诈性破产.他们引入了一种社交网络结构,从而基于社交网络的挖掘算法来进行整个欺诈集团的挖掘.在软件欺诈检测方面,Zhu 等人[9]进行了移动应用软件的排名欺诈检测,排名欺诈的目的是提升应用软件在流行度列表中的排名.在软件恶意破坏检测方面,Heindorf 等人[10]提出了维基百科编辑恶意破坏行为检测.Kumar 等人[11]设计了维基百科恶意破坏预警系统.

在图 2 的子过程(b)中,采用了数据仓库中数据方体的概念,但是通常数据仓库的方体操作需要求其不能是稀疏的,即:每个方体中都应该有数据,并且方体中只能进行简单的聚合操作.为了克服稀疏性,并且采用人工智能和机器学习技术来对方体中的异常数据进行深入的分析.本文在数据方体基础上提出了数据块的概念,这是子过程(b)和 OLAP 技术的最大区别.在基于 OLAP 的异常检测方面,李等人[12]提出了基于时间序列的数据立方体来构建异常分析的多维空间,Henrion 等人[13]提出一种高维天文学数据库中异常检星系的检测方法,Heine 等人[14]基于 OLAP 技术从流数据中进行异常值检测,Dalmia 等人[15]试图从由图模型转换而来的数据立方体中进行异常值检测.虽然 OLAP 方法是通用的,但是由不同种类的数据所构造成的数据立方体各不相同地针对不同的原始数据,需要设计不同的算法来将其转换为多维立方体.另一方面,由于多维异常检测通常需要找到前 k 个最可能是异常的数据立方体,许多算法致力于设计不同的数据立方体异常度(local outlier factor,简称 LOF)的设计.Kriegel 等人[16]基于距离的相似性计算不同子空间的 LOF.He 等人[17]、Vreeken 等人[18]提出了基于频繁项集的 LOF 计算方法.Muller 等人[19]通过分析不同子空间递归关系计算每个子空间的 LOF.

在图 2 的子过程(c)~(h)中,我们主要采用了基于张量分解进行异常检测的方法.从张量分析方法上和其他采用张量进行异常分析的方法是比较相似的,最大的不同是构造张量的过程.通常采用张量进行异常分析的方法其张量本身的构建是比较直接的,比如基于社交网络的关联关系,同时结合时间维度构造张量,或基于用户与其发表评论的关键词,同时结合时间维度构造张量.比如,Jiang 等人[20]基于复杂网络 SVD 分解来检测假的网络名人和粉丝,他们还提出了一种通用度量和算法[21]来检测多模态行为数据中具有可疑行为的密集子图.Eswaran 等人[22]通过构建用户信任传播网络,发现了电子商务门户中的欺诈交易行为.Costa 等人[23]提出了一种通用的休息-睡眠-评论模型,以匹配用户在社交媒体上发布评论的模式.本文是通过数据块的特征提取来构造张量.二者具有很大的不同.由于欺诈事件的复杂性,很难设计出通用的张量构造方法.

通过上述工作分析可知:要解决本文特定的挂单欺诈问题,很难使用现有研究工作的成果直接进行问题的

求解,必须融合多种技术,如数据仓库、人工智能、机器学习等技术,综合起来形成一套专门求解本文问题的新的求解框架,这正是本文的主要贡献.

3 特定挂单模式下的挂单点挖掘

3.1 挂单点特征提取

为了挖掘挂单模式下的真实挂单点,需要提取出挂单点的特征,图 3 绘制了销售数据仓库中所有销售额的分布图,横坐标表示销售额,纵坐标表示该销售额区间的出现频率,横纵坐标都转换成了双对数坐标.由于在双对数坐标下,分布图趋近于直线,所以推测销售额呈幂律分布.因此,可以将每个挂单点销售额序列的幂律分布参数作为该挂单点的数据特征.但由于数据的稀疏性会使得某些挂单点只包含很少销售记录,导致无法计分布参数,本文设计了分割率来替代幂律分布参数以克服数据稀疏性,这里首先给出相关定义.

定义 8(分割率 *ratio*($p_{C_{i,s}}$)). 设有挂单点 $p_{C_{i,s}}=\{t_1,...,t_N\}$,$\{v_1,...,v_N\}$是$\{t_1,...,t_N\}$中每条记录所含销售额的降序序列,对任意 $v_i\in\{v_1,...,v_N\}$,有 $ratio_i=mean(v_1,...,v_i)/mean(v_{i+1},...,v_N)$,其中,$mean(v_1,...,v_i)$和 $mean(v_{i+1},...,v_N)$分别表示 $v_1,...,v_i$ 和 $v_{i+1},...,v_n$ 的均值,$ratio(p_{C_{i,s}})=\max(ratio_1,...,ratio_{n-1})$.

定义 9(挂单点的头部平均值 $H(p_{C_{i,s}})$ 和尾部平均值 $T(p_{C_{i,s}})$). 设有挂单点 $p_{C_{i,s}}=\{t_1,...,t_n\}$,则 $H(p_{C_{i,s}})=mean(v_1,...,v_i)$,$T(p_{C_{i,s}})=mean(v_{i+1},...,v_N)$,其中,$i$ 是使得 $ratio_i$ 是$\{ratio_1,...,ratio_n\}$中最大的那个点.

设每个挂单点 $p_{C_{i,s}}$ 包含的销售记录服从幂律分布 $P(v>x)=(x_{\min}/x)^{\alpha}$,其中,$x_{\min}$ 和α分别是幂律分布的两个参数.图 4 展示了每个挂单点的 *ratio* 和 $x_{\min}$ 之间的关系,其中,$x_{\min}$ 通过对挂单点 $p_{C_{i,s}}$ 的销售额序列进行极大似然估计得到,*ratio* 通过对挂单点销售额序列按照定义 8 计算得到.直线是 *ratio* 和 $x_{\min}$ 的线性拟合结果,可以看出,散点图较好的符合了该直线.这说明可以将 *ratio* 作为 $x_{\min}$ 的近似;同时,由于 *ratio* 只需要销售额序列包含 2 个及以上的数据,所以解决了稀疏数据特征提取的问题.

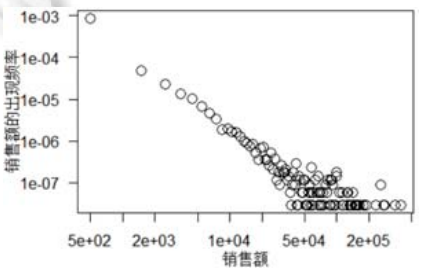


Fig.3 Distribution of sale volumes

图 3 销售额分布

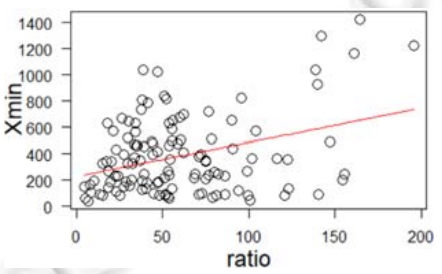


Fig.4 Relation between *ratio* and α

图 4 *ratio* 和α的相关关系

例 2(不同提取方法得到的挂单点特征值):图 2(c)展示了图 2(a)在两种挂单模式下得到两种挂单点集合,图 2(d)展示了在挂单模式(商品类型,分销商类型,*)下,采用 3 种不同方法提取出的挂单点特征.$x_{\min}$ 表示基于幂律分布的最大似然估计得到每个挂单点的 $x_{\min}$ 值,*ratio* 表示基于定义 7 得到每个挂单点的分割率,*mean* 表示直接求每个挂单点包含销售额集合的平均值,这是 OLAP 中使用较多的聚集函数.可以看到,部分挂单点无法得到 $x_{\min}$.虽然 *mean* 和 *ratio* 都能得到计算结果,*ratio* 明显更能体现真实挂单点与其他挂单点的差异.

3.2 挂单点候选集过滤

由于真实挂单点肯定含有较大销售额,也即肯定是具有较大的头部平均值,则在计算挂单点权重之前,应该将那些头尾平均值都较小的挂单点过滤掉.同时,第 3.1 节指出挂单点所包含的销售额序列呈幂律分布,所以过滤过程需要在不同规模尺度的头尾部平均值上逐层迭代进行,算法 1 给出了具体过程.

算法 1. 层次化挂单点候选集过滤算法 *H_Filter*(*l*,*P*).

输入:挂单模式 *l* 下的挂单点候选集 $P=\{p_{C_{1,s}},p_{C_{2,s}},...,p_{C_{k,s}}\}$,过滤算法迭代次数 *k*;

输出:*P*′=过滤后的挂单点候选集.

1. 初始化 *P*′=∅, *i*=1
2. **WHILE** *i*<*k*
3. $P=\{\{H(p_{C_{1,s}}),T(p_{C_{1,s}})\},...,\{H(p_{C_{k,s}}),T(p_{C_{k,s}})\}\}$;
4. $(P_1,P_2)=GMM\_EM(P)$;
5. *i*=argmax(*avg*(*H*(P_i))); //*i*=1, 2
6. $P'=P'\cup P_i$;
7. $P=P-P'$;
8. *k*=*k*+1;
9. **END WHILE**

算法第 3 行求出每个挂单点的头尾部平均值作为每个挂单点的特征值;第 4 行基于混合高斯模型聚类算法对 *P* 进行 2 分聚类;第 3 行~第 9 行进行 *k* 次循环,将每次将聚类结果中头部均值较大的一类保留到最终的候选集 *P*′中,将头部均值较小的一类作为下一次聚类的输入数据.

例 3(挂单点候选集过滤示例):图 3(e)描述了基于 *ratio* 和 *mean* 两种挂单点特征进行过滤的结果.*P* 是图 2(d)中的所有挂单点.在 *ratio* 特征下:

- 首先对 *P* 进行 2 分聚类得 $P_1=\{p_{(美妆店,飘柔,*),s_2}\}$,$P_2=P-P_1$.设 $H(P_1)>H(P_2)$,则 $P'=\{p_{(美妆店,飘柔,*),s_2}\}$,$P=P_2$;
- 继续对 *P* 进行 2 分聚类得 $P_1=\{p_{(批发市场,飘柔,*),s_3},p_{(批发市场,潘婷,*),s_3},p_{(批发市场,飘柔,*),s_4}\}$,$P_2=P-P_1$.设 $H(P_1)>H(P_2)$,则 $P'=P_1$.

如果循环次数 *k*=2,则算法停止,过滤后的挂单点候选集:

$$P'=\{p_{(美妆店,飘柔,*),s_2},p_{(批发市场,飘柔,*),s_3},p_{(批发市场,潘婷,*),s_3},p_{(批发市场,飘柔,*),s_4}\}.$$

而在 *mean* 特征下,$p_{(批发市场,飘柔,*),s_4}=1.2$,值很小,被算法 1 过滤掉了.

3.3 计算挂单点的异常度

由于过滤后的挂单点候选集中各个挂单点的销售都相对较大,无法通过销售额绝对值来判断挂单点异常性,只能通过挂单点所体现的销售行为异常性来进行区分.本节首先根据过滤后挂单点候选集及其挂单模式构造张量,然后通过张量分解和重构来计算挂单点异常度.

算法 2. 挂单点异常度计算 *Outlying_degree*(*l*,*P*′).

输入:*l*=(a_1,a_2,…,a_n),通过算法 1 过滤后的挂单点候选集 *P*′,*n*+1 阶张量 *A*,*A*=(*s*,a_1,a_2,...,a_n),*s* 维的长度是 *P*′中所有分销商的数量,a_i 维的长度是 *P*′中所有挂单点在 a_i 维上的取值数量;

输出:*A*(*P*′),即 *P*′中每个挂单点的异常度.

1. **FOR EACH** $p_{C_{i,s}}$ **IN** *P*′
2. $A_{s_i,a_1,a_2,...,a_n}=1$; //其中,$s_i$=*s*,$a_1$,$a_2$,...,$a_n$ 是 C_i 在挂单模式 *l* 上的具体取值
3. **END FOR**
4. **FOR** *i* in (*s*,a_1,a_2,...,a_n)
5. $A_i=Unfold(A,i)$;
6. **END FOR**
7. **FOR** *i* in (*s*,a_1,a_2,...,a_n)
8. $A_i=SVD(A_i)=U^{(i)}\times S_i\times V_i^T$;
9. **END FOR**

10. $S = A \times U_{\lambda s}^{T} \times U_{\lambda a_1}^{T} \times U_{\lambda a_2}^{T} \times ... \times U_{\lambda a_n}^{T}$;
11. $A' = S \times U_{\lambda s} \times U_{\lambda a_1} \times U_{\lambda a_2} \times ... \times U_{\lambda a_n}$;
12. $A(P') = A'_{\mathrm{MAX}} - A'$;

下面对算法 2 稍作解释.

(1) 算法 2 的第 1 行~第 3 行首先构造 n+1 维张量,$(a_1,a_2,...,a_n)$维表示挂单模式 l=$(a_1,a_2,\ldots,a_n)$,s 维表示分销商 ID.同时,如果$(a_1,a_2,\ldots,a_n)$中存在 a_i=*时,则应该在第 1 步中去掉该维,因为当 a_i=*时,此维度的长度为 0,无法构造张量,此时 l=$(a_1,a_2,\ldots,a_{i-1},a_{i+1},...,a_n)$.比如图 2(e)中,在 *ratio* 特征下得到过滤后的挂单点候选集 $P' = \{p_{(\text{美妆店,飘柔,*}),s_2}, p_{(\text{批发市场,飘柔,*}),s_3}, p_{(\text{批发市场,潘婷,*}),s_3}, p_{(\text{批发市场,飘柔,*}),s_4}\}$,$P'$的挂单模式为 l=(分销商类型,商品系列,*),则根据算法 2 构造的张量维度为(分销商 ID,分销商类型,商品系列).可以看出,P'中的销售行为可以分为两类:第 1 类销售行为中分销商属于批发市场,并且对飘柔和潘婷系列的销售额都较高;第 2 类销售行为中分销商属于美妆店,并且只对飘柔系列的销售额较高.第 1 类销售行为占据了候选集中的 3/4 记录数据,第 2 类占据 1/4;

(2) 第 4 行~第 6 行表示将 A 在每个模式上分别进行展开;

(3) 第 7 行~第 9 行表示对展开后的矩阵 $A_s, A_{a_1}, A_{a_2}, ..., A_n$ 进行 SVD 分解;

(4) 第 10 行表示对 A 约减得到核心张量 S;

(5) 第 11 行表示通过核心张量 S 重构约减后的张量 A'.A'体现了挂单模式中的主要销售行为,如图 5(f)所示.可知第 1 类销售行为中的 3 条销售记录在 A'中均有较大值,所以可以推断第 1 类销售行为体现了该销售数据的主要销售行为;而第 2 类销售行为中的挂单点 $p_{(\text{美妆店,飘柔,*}),s_2}$ 在 A'中取值为 0,说明其销售行为并非主流行为,很可能是真实挂单点;

(6) 第 6 行中的 A'_{MAX} 表示将 A'中的最大值作为 A'中每个元素的值,二者相减后,可以使得越异常的元素的值越大,如图 2(g)所示. $\lambda_s, \lambda_{a_1}, ..., \lambda_n$ 是对第 4 行~第 6 行展开后的矩阵分别进行奇异值分解的参数,它们决定了分解后所保留的主成份个数,在实验中, $\lambda_s, \lambda_{a_1}, ..., \lambda_n$ 的取值为其中的最小值.对比图 2(g)可以看到:真实挂单点 $p_{(\text{美妆店,飘柔,*}),s_2}$ 在特性 *ratio* 下具有非常明显的异常性;而在特征 *mean* 下,真实挂单点 $p_{(\text{美妆店,飘柔,*}),s_2}$ 和非真实挂单点 $p_{(\text{批发市场,潘婷,*}),s_3}$ 具有同样高的异常度,这就降低了检测的准确性.

4 挂单模式挖掘

4.1 基于挂单点候选集异常度分布的挂单模式特征提取

挂单模式挖掘旨在判断出销售数据所服从的挂单模式是所有候选挂单模式的哪一种,是典型的分类问题,所以需要从数据中进行分类特征的提取.由于数据本身存在稀疏性和正常极值问题,所以直接从原始数据中进行特征提取是比较困难的.通过大量实验发现:如果原始数据仓库中的真实挂单模式为 l,那么无论在算法 2 中输入哪种挂单模式进行分析,其输出的挂单点集合异常值分布都是相对比较相似的.因此,我们将基于原始数据进行异常值计算后的结果进行特征提取,具体见算法 3.

算法 3. 基于异常度的挂单模式分类特征提取算法 *Basic_feature_extraction*(R).

输入:含挂单行为的销售数据仓库 R,R 所包含的挂单模式集合$\{l_1,l_2,\ldots,l_q\}$,分箱宽度 b;

输出:R 的特征属性向量 v(R).

1. 初始化 $v(R)=\varnothing$;
2. **FOR** i in (1,2,...,q)
3. $P(l_i)$=*Outlying_degree*(l_i,*H_Filter*(l_i,R));
4. $v(R)=v(R)\cup$将 $P(l_i)$进行 b 等分等宽分箱;
5. **END FOR**

第 2 行表示共有 q 个挂单模式.第 3 行表示先通过算法 1 在挂单模式 l_i 下进行 R 的挂单点候选集过滤,再

通过算法 2 在挂单模式 l_i 下计算过滤后挂单点候选集中各挂单点的异常值,$P(l_i)$表示异常值的集合.第 4 行表示对 $P(l_i)$进行 b 等分等宽分箱,一共可以构造 $q*b$ 个特征属性.比如,设 $P(l_i)$=(0.1,0.1,0.8,0.8,1.2),按照 b=(0,0.5)和(0.5,以上)进行 2 等分等宽分箱,则 R 在 l_i 下可以得到两个特征属性(2,3),因为 $P(l_i)$中 0~0.5 之间的异常值有 2 个,0.5 以上的异常值有 3 个.如果总共有 q=7 个挂单模式,则通过算法 3 对 R 构造的特征向量一共含有 2×7=14 个特征属性.

4.2 基于挂单点异常度分布和挂单模式偏序结构的挂单模式特征提取

经过大量实验发现,在第 4.1 节得到的分类特征基础上加入挂单模式偏序结构信息,将会提高挂单模式分类算法的分类精度.

定义 9(父子挂单模式). 设有销售数据仓库 R,已知 R 中多维数据空间 D 上的偏序格 $L=\langle M,\preccurlyeq\rangle$,$M=\{l_1,...,l_n\}$,设 M 中有任意两个挂单模式 l 和 l',$l=(a_1,a_2,\ldots,a_n)$, $l'=(a_1',a_2',...,a_n')$,如果存在且只存在一个维度 $i(1\leqslant i\leqslant n)$,有 a_i' 是 a_i 的上一个概念级别,则称 l'是 l 的父挂单模式,记为 $l\preccurlyeq_p l'$.

算法 4. 基于挂单模式偏序格的挂单模式分类特征提取算法 *Advanced_feature_extraction*(R).

输入:含挂单行为的销售数据仓库 R,R 所包含的挂单模式集合$\{l_1,l_2,\ldots,l_q\}$,分箱宽度 b;

输出:R 的特征属性向量 $v(R)$.

1. 初始化 $v(R)=\varnothing$, $P=\varnothing$
2. **FOR** i in (1,2,...,q)
3. **FOR** j in (1,2,...,q)
4. **IF** $l_i\preccurlyeq_p l_j$;
5. **THEN**
6. $P(l_i)$=*Outlying_degree*(l_i,*H_Filter*(l_i,R));
7. $P(l_j)$=*Outlying_degree*(l_j,*H_Filter*(l_j,R));
8. $P=P(l_i)\cup P(l_j)$;
9. $v(R)=v(R)\cup$将 P 进行 b 等分等宽分箱;
10. $P=\varnothing$;
11. **END IF**
12. **END FOR**
13. **END FOR**

第 6 行~第 10 行旨在计算 R 在挂单模式偏序格上具有父子关系两个挂单模式的异常度,再将其组合后进行分箱处理,过程与算法 3 相同.然后,基于父子挂单模式构造 R 的特征向量.如图 1(b)所示,多维数据空间 D={分销商,商品}上的偏序格为 $L=\langle M,\preccurlyeq\rangle$,$L$ 中共有 10 组父子挂单模式,对应图 1(b)上的 10 条边,去掉(*,*)所对应的两组父子挂单模式,共有 8 组父子挂单模式,则 $v(R)$共含有 2×8=16 个特征属性.

设有 n 个挂单模式已知的销售数据集$\{(R_1,l_1),...,(R_n,l_n)\}$,$R_i$ 表示第 i 个数据集,l_i 表示其真实挂单模式,采用算法 3 和算法 4 中的方法可提取 R_i 的特征向量 $v(R_i)$,得到分类算法训练集$\{(v(R_1),l_1),...,(v(R_n),l_n)\}$,采用分类算法训练后可对未知挂单模式的数据集进行挂单模式分类,本文采用随机森林[24]作为分类算法.

5 实验结果与分析

原始数据总共包含 362 045 条销售记录,分为两种类型:第 1 种是没有任何挂单行为的干净数据 R_{clean},共 200 000 条数据;第 2 种是存在着挂单行为的数据 R_{real},共 162 045 条数据.原始数据包含 3 个维度{时间,商店,商品},每个维度的层次关系为:月份→年→All,商品 ID→商品系列→商品品牌→商品类型→All,分销商 ID→分销商类型→All.为验证算法的有效性,我们在真实数据和通过真实数据进行模拟后的合成数据上进行了实验,整个算

法基于 R 语言编写,所有实验均在 Windows 7,16G 内存和 Inter i7 的运行环境下完成.

5.1 在合成数据上的分析

合成数据通过对 R_{clean} 进行模拟挂单得到,下面简单介绍模拟挂单过程.设 $R_{clean}=\{t_1,t_2,...,t_N\}$,挂单模式为 l,抽样记录数为 s,抽样阈值为 p.首先,随机从 R_{clean} 中选出 s 条销售额大于 p 的记录 S,$S=\{t_1,t_2,\ldots,t_s\}$;设 $Chunk(R_{clean},l)=\{C_1,C_2,\ldots,C_k\}$,并且任意 $C_i(1\leqslant i\leqslant k)$只包含 S 中的记录;最后,对任意 C_i,随机选取 C_i 中一条记录将其销售额变成 C_i 中所有销售额的总和,将 C_i 中其他记录的销售额变成 0.在具体实验中,我们挑选了图 1(b)中的 7 个挂单模式作为挂单模式的选择范围,时间维度均为*,见表 1.

Table 1 Sale accumulation pattern and abbreviation

表 1 挂单模式及简称

简称	挂单模式
C	(商品类别,*,*)
B	(商品品牌,*,*)
V	(商品系列,*,*)
CT	(商品类别,分销商类型,*)
BT	(商品品牌,分销商类型,*)
VT	(商品系列,分销商类型,*)
T	(*,分销商类型,*)

本节的实验分析了 3 个指标.

1) 在已知挂单模式的情况下,算法 1 得到的候选挂单点集合对真实挂单点集合的覆盖率.

算法 1 采用了本文提出的 *ratio* 指标来提取挂单点的特征,为了进行对比,将 *ratio* 指标分别替换成基于频率分布的参数(简记为 *freq*)和基于幂律分布的参数(简记为 $x_{\min}$).其中,*freq* 特征统计每个候选挂单点所包含的销售额集合在 3 个取值区间——(100000 以上)、(100000,10000)、(10000 以下)上的记录个数,$x_{\min}$ 特征用极大似然估计计算每个候选挂单点销售额序列所服从的离散幂律分布的参数 $x_{\min}$.表 2 给出了实验结果,其中,α表示通过模拟得到的真实挂单点个数,β表示通过算法 1 得到的候选挂单点集合大小,$\cap$表示α和β的交集元素个数.每个实验结果均是将模拟过程的抽样记录数 s 分别设为(50,100,150,200)的 4 次实验结果的平均值.相同的迭代次数下,*ratio* 得到的挂单点候选集是最小的,但包含真实挂单点数目却是最多的.基于 *ratio*,算法 1 迭代 4 次后,候选集就能包含所有的真实挂单点.基于 $x_{\min}$ 和 *dis*,算法 1 得到候选集较大,且迭代 4 次后不能包含所有真实挂单点.

Table 2 Coverage rate of algorithm 1 on the true sale accumulation points

表 2 算法 1 的输出对真实挂单点集合的覆盖率

		(*ratio*,2)		(*ratio*,3)		(*ratio*,4)		($x_{\min}$,2)		($x_{\min}$,3)		($x_{\min}$,4)		(*freq*,2)		(*freq*,3)		(*freq*,4)	
	α	β	$\cap$	β	$\cap$	β	$\cap$	β	$\cap$	β	$\cap$	β	$\cap$	β	$\cap$	β	$\cap$	β	$\cap$
C	3	41	3	42	3	155	3	44	2	130	3	260	3	51	2	158	3	289	3
B	8	9	4	54	6	176	8	76	3	187	4	357	5	52	2	235	3	290	3
V	10	8	3	68	9	258	10	123	6	270	7	418	8	160	4	217	7	459	8
CT	7	6	3	39	6	157	7	78	4	164	5	292	6	92	4	213	4	388	5
BT	11	9	3	42	8	157	11	131	6	245	9	408	10	250	5	314	7	511	7
VT	13	6	2	41	9	164	13	128	5	293	9	476	9	157	3	281	6	569	7
T	6	6	4	46	6	105	6	52	2	153	5	327	6	113	2	272	3	481	4

2) 算法 2 挖掘的精度.

本节通过绘制 *ROC* 曲线来分析算法 2 的挖掘效果,*ROC* 曲线绘制公式为

$$TPR_p=\frac{\text{异常值大于等于}A(p)\text{的真实挂单点个数}}{\text{真实挂单点个数}},$$

$$FPR_p=\frac{\text{异常值大于等于}A(p)\text{的虚假挂单点个数}}{\text{虚假挂单点个数}}.$$

真实挂单点指该挂单点中包含有挂单记录,虚假挂单点指该挂单点中不包含挂单记录,$A(p)$为通过算法 2 得到 p 的异常值.由于算法 2 需要输入挂单点候选集和挂单模式,本节进行两方面的实验对比.

首先,对比了挂单模式不变而挂单点候选集改变时的挖掘精度.如表 3 第 1 行第 1 列 *AUC* 值为 0.65,*C* 表示算法 2 输入的挂单模式和真实挂单模式均为 *C*,(*ratio*,4)表示算法 2 采用 *ratio* 作为特征提取方法, 4 表示算法 1 的迭代次数.当算法 1 采用 *ratio* 特征进行候选集过滤时,算法 2 能够达到 0.65 的平均 *AUC* 值,远高于采用 $x_{\min}$ 和 *freq* 进行特征提取得到的 0.36 和 0.3;

其次,对比了候选集不变而挂单模式改变时的挖掘精度.图 5 中每张图的标题分别表示真实挂单模式,算法 2 中输入的挂单模式和算法 2 结果 *ROC* 曲线的 *AUC* 值.每张图中包含 10 条 *ROC* 曲线,是将模拟过程的抽样记录数 *s* 设为 100 时的 10 次实验结果.由于篇幅原因,只画出表 1 中的 3 种挂单模式 *V*,*CT*,*BT* 的输出结果.

Table 3 *AUC* value of algorithm 2 with different sale accumulation point candidate set

表 3 算法 2 输入不同挂单点候选集时的 *AUC* 值

	C	*B*	*V*	*CT*	*BT*	*VT*	*T*
ratio,4	0.65	0.58	0.64	0.68	0.64	0.64	0.74
$x_{\min}$,4	0.36	0.33	0.34	0.14	0.28	0.25	0.07
dis,4	0.3	0.28	0.26	0.07	0.2	0.21	0.03

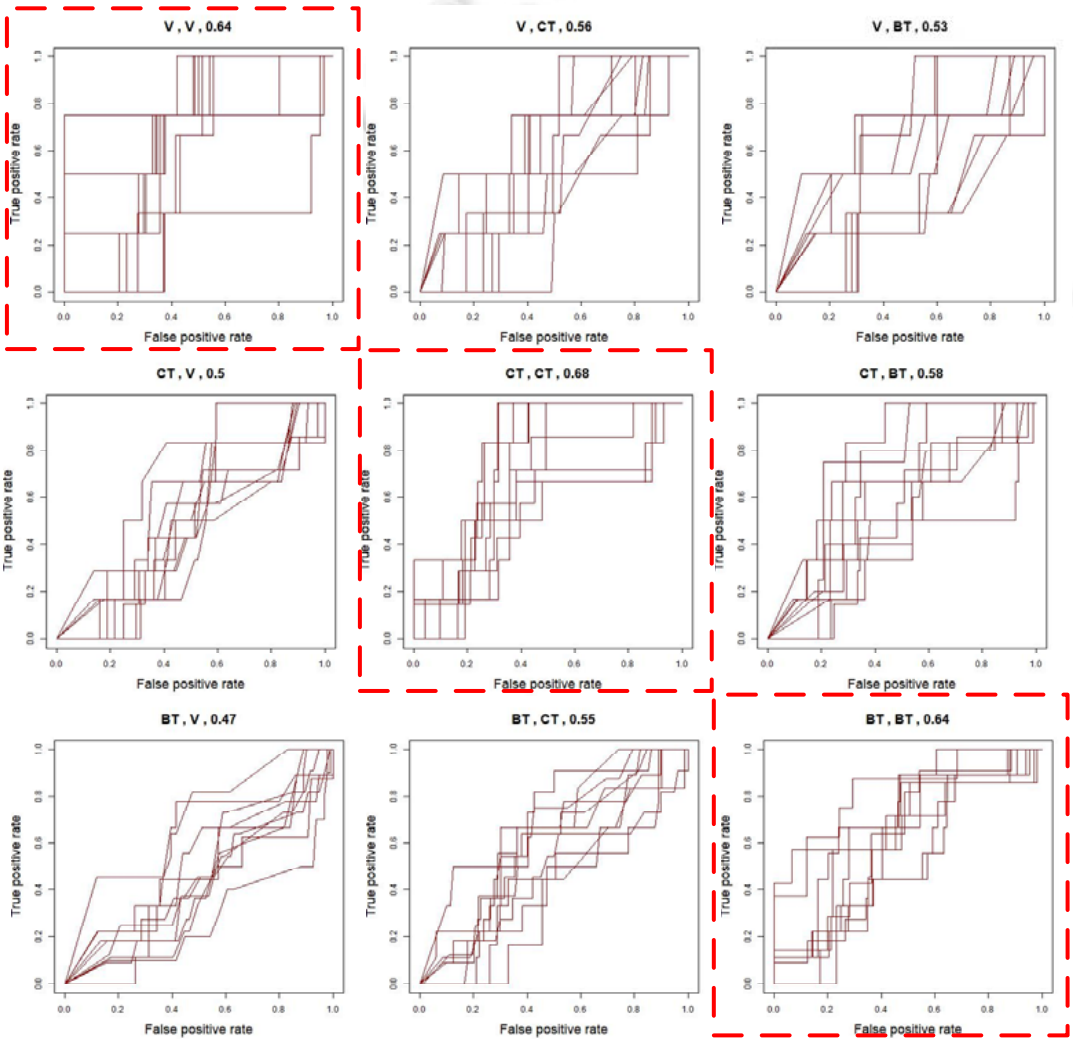


Fig.5 Precision of algorithm 2 when candidate set is unchanged and the sale accumulation pattern is changing

图 5 算法 2 候选集不变而挂单模式改变时的挖掘精度

可以看到每一行的 *ROC* 曲线是比较相似的,即算法 2 输入不同挂单模式进行求解得到的 *ROC* 曲线是相似的,这表明了第 4.1 节通过算法 2 在不同挂单模式下的挖掘结果来反推真实挂单模式的可行性;其次,虚线框表示在每一行中相对较大的 *AUC* 值,及算法 2 输入的是真实挂单模式或者真实挂单模式的父子模式,则可以得到相对较高的挖掘精度,这表明了第 4.2 节中通过挂单模式偏序结构来反推真实挂单模式的可行性.

3) 挂单模式分类精度.

采用第 5.1 节所给出的方法进行多次模拟实验来得到不同的合成数据,其中,l 的取值范围为表 2 中所给出的 7 种挂单模式,s 的取值范围为(50,100,150,200),p 固定设置为 10 000.每种组合进行 5 次模拟采样,则最终得到的数据集 S 包含 7×4×5=140 条数据,即 $S=\{(R_1,l_1),...,(R_{140},l_{140})\}$,$R_i$ 表示模拟挂单后得到的合成数据集,l_i 表示该数据集的真实挂单模式.通过算法 3 和算法 4 节给出的方法在 S 中的每个 R_i 上进行特征提取,得到 140 条训练数据,然后采用随机森林分类算法进行分类,表 4 给出了在 S 上将 70 条数据作为训练数据,将另外 70 条数据作为测试数据,抽样记录数 s 分别取(50,100,150,200)时的实验结果.可以看到,采用基于算法 4 提取的特征得到的分类精度明显优于算法 3.

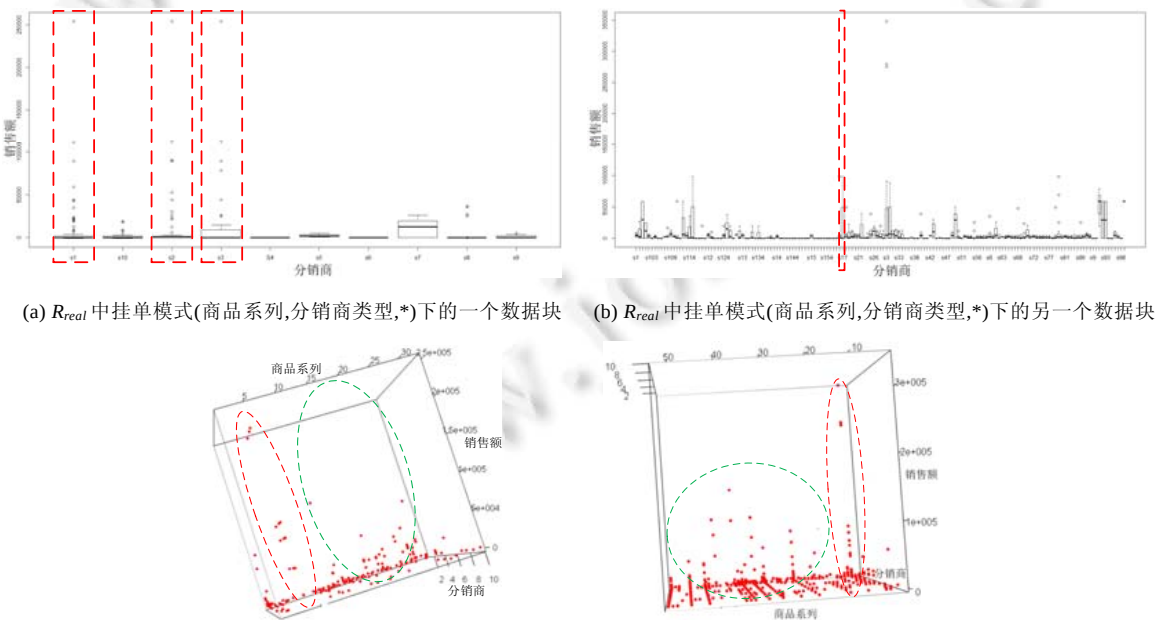
Table 4 Sale accumulation classification precision via different feature extraction method

表 4 基于不同特征提取方法的挂单模式分类精度

	50	100	150	200
算法 3 (%)	27.2	35.1	20	24.2
算法 4 (%)	34.3	37.4	40	37.2

5.2 在真实数据上的分析

R_{real} 共包含 162 045 条销售记录,总共包含 977 家分销商,1 542 种商品,从 2014 年 7 月~2015 年 8 月.采用算法 4 判断出该数据集的挂单模式为(商品系列,分销商类型,*).在该挂单模式下,通过算法 1 和算法 2 进行真实挂单点挖掘,图 6(a)~图 6(d)展示了部分挖掘结果.



(a) R_{real} 中挂单模式(商品系列,分销商类型,*)下的一个数据块 (b) R_{real} 中挂单模式(商品系列,分销商类型,*)下的另一个数据块

(c) 图 6(a)的数据块中所有的商品销售记录 (d) 图 6(b)的数据块中所有的商品销售记录

Fig.6 Analyzing results on the data set with real sale accumulation behavior

图 6 在存在真实挂单行为数据上的分析结果

图 6(a)、图 6(b)表示该挂单模式下的一个数据块,图中每一竖列表示数据块中的一个挂单点,横坐标表示挂单点对应的分销商,纵坐标表示挂单点包含的销售额集合对应的箱线图,其上端圆圈是箱线图无法拟合的异常点,红色虚线框中是具有较高异常值的挂单点.首先可以看出,在图 6(a)、图 6(b)中存在大量箱线图无法拟合的极值点,但本文的算法并没有对所有极值点所在的候选挂单点都给出很高的异常值,这说明本文的算法能够区分正常极值点和异常极值点.图 6(c)、图 6(d)是图 6(a)、图 6(b)表示的数据块中所有的商品销售记录.图 6(c)、图 6(d)中,红色圆圈内的销售记录对应图 6(a)、图 6(b)红色虚线框中的极值点;图 6(c)、图 6(d)中的绿色圆圈对应图 6(a)、图 6(b)中其他没在红色圆圈中的极值点.由于图 6(c)、图 6(d)中绿色圆圈中的极值点呈现出聚类的趋势,说明本文算法能从主流销售行为角度区分正常极值点和异常极值点.

5.3 算法的性能及时间复杂度分析

表 5、表 6 给出了在输入不同数量挂单点情况下算法 1、算法 2 的平均运行时间.可以看出,算法 1、算法 2 基本上呈线性复杂度增长.由于算法 2 总是在算法 1 的基础上运行,故表 6 中的挂单点数量总是远小于表 5.

Table 5 Average running time of algorithm 1 with different number of sale accumulation points (s)

表 5 算法 1 在不同数量挂单点输入下的平均运行时间 (s)

挂单点	877	1 596	3 279	6 839
算法 1	3	6	14	35

Table 6 Average running time of algorithm 2 with different number of sale accumulation points (s)

表 6 算法 2 在不同数量挂单点输入下的平均运行时间 (s)

挂单点	21	69	108	152
算法 2	2	6	16	39

设销售数据仓库 R 上的挂单模式偏序格为 $L=\langle M,\preccurlyeq\rangle$,则算法 3 的时间复杂度为 $O(T\times|M|)$,算法 4 的时间复杂度为 $O(T\times|M\times M|)$,其中,T 是算法 1 和算法 2 联合运行的时间,$|M|$是 M 中挂单模式的数量,$|M\times M|$是 M 中父子挂单模式的数量,故算法 3、算法 4 与算法 1、算法 2 的运行时间成正比.

6 总结及今后的工作

本文提出的挂单点和挂单模式挖掘能够分析出分销渠道挂单的方式和具体挂单的销售记录,能够帮助审计部门在大数据场景下快速进行挂单欺诈行为分析.针对销售数据特有的正常极值和数据稀疏性问题,本文提出了基于分割率的特征提取方法和基于张量重构的挂单行为挖掘算法来挖掘挂单点,提出了基于挂单模式偏序格的特征提取方法来挖掘挂单模式,在合成数据和真实数据的实验中,验证了本文提出方法的有效性.

在今后的工作中,将从如下 3 个方面进行改进.首先,将继续改进挂单模式挖掘的特征提取方法,从而提高挂单模式分类的精度;其次,将尝试把基于有监督方式的挂单模式挖掘改为基于无监督方式的挖掘,这将降低算法对数据的要求;最后,将尝试在本文问题 2 所提出的第 1 种尺度上进行挂单点挖掘,这将进一步提高欺诈行为检测的精度.

References:

[1] Kenneth G, Magrath AJ. Dealing with cheating in distribution. European Journal of Marketing, 1989,23(2):123−129. [doi: 10.1108/eum0000000000551]

[2] Shu K, Luo P, Li W, Yin F, Tang L. Deal or deceit: Detecting cheating in distribution channels. In: Proc. of the 23rd ACM CIKM Int'l Conf. on Information and Knowledge Management. San Francisco: ACM, 2013. 1419−1428. [doi: 10.1145/2661829.2661874]

[3] Zhang R, Zheng F. Sequential behavioral data processing using deep learning and the Markov transition field in online fraud detection. In: Proc. of the 24th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. London: ACM, 2018. 1−5. [doi: 10.1093/obo/9780199828340-0063]

[4] De Roux D, Perez B, Moreno A, Villamil MDP Figueroa C. Tax fraud detection for under-reporting declarations using an unsupervised machine learning approach. In: Proc. of the 24th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. London: ACM, 2018. 215−222.

[5] Jiang ST, Min W, Gao Q. Question and answer feature extracting framework for online lending collection risk modeling with xencoder. In: Proc. of the 11th ACM WSDM Int'l Conf. on Web Search and Data Mining. Los Angeles: ACM, 2018. 1211−1215.

[6] Min W, Tang ZY, Zhu M, Dai YX, Wei Y, Zhang RN. Behavior language processing with graph based feature generation for fraud detection in online lending. In: Proc. of the 11th ACM WSDM Int'l Conf. on Web Search and Data Mining. Los Angeles: ACM, 2018. 1430−1436.

[7] Vlasselaer V, Eliassi-Rad T, Akoglu L, Snoeck M, Baesens B, Afraid: Fraud detection via active inference in time-evolving social networks. In: Proc. of the 11th ACM ASONAM Int'l Conf. on Advances in Social Networks Analysis and Mining. Paris: ACM, 2015. 659−666. [doi: 10.1145/2808797.2810058]

[8] Vlasselaer V, Akoglu L, Eliassi-Rad T, Snoeck M. Guilt-by-constellation: Fraud detection by suspicious clique memberships. In: Proc. of the 48th IEEE HICSS Hawaii Int'l Conf. on System Sciences. Hawaii: IEEE, 2015. 918−927. [doi: 10.1109/hicss.2015.114]

[9] Zhu H, Xiong H, Ge Y, Chen E. Discovery of ranking fraud for mobile apps. IEEE Trans. on Knowledge and Data Engineering, 2015,27(1):74−87. [doi: 10.1109/TKDE.2014.2320733]

[10] Heindorf S, Potthast M, Stein B, Engels G. Vandalism detection in wikidata. In: Proc. of the 25th ACM CIKM Int'l on Conf. on Information and Knowledge Management. Indiana: ACM, 2016. 327−336. [doi: 10.1145/2983323.2983740]

[11] Kumar S, Spezzano F, Subrahmanian V. VEWS: A Wikipedia vandal early warning system. In: Proc. of the 21st ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. Sydney: ACM, 2015. 607−616. [doi: 10.1145/2783258.2783367]

[12] Li X, Han J. Mining approximate top-*k* subspace anomalies in multi-dimensional time-series data. In: Proc. of the 33rd ACM VLDB Int'l Conf. on Very Large Data Bases. Vienna: ACM, 2007. 447−458. [doi: 10.1023/A:1015417610840]

[13] Henrion M, Hand D, Gandy A, Mortlock D. Casos: A subspace method for anomaly detection in high dimensional astronomical databases. Statistical Analysis and Data Mining the Asa Data Science Journal, 2013,6(1):53−72. [doi: 10.1002/sam.11167]

[14] Heine F. Outlier detection in data streams using olap cubes. In: Proc. of the Communications in Computer and Information Science. 2017. 29−36. [doi: 10.1007/978-3-319-67162-8_4]

[15] Dalmia A, Gupta M, Varma V. Query-based graph cuboid outlier detection. In: Proc. of the 11th ACM ASONAM Int'l Conf. on Advances in Social Networks Analysis and Mining. Paris: ACM, 2015. 101−113. [doi: 10.1145/2808797.2810061]

[16] Kriegel H, Kroger P, Schubert E, Zimek A. Outlier detection in axis-parallel subspaces of high dimensional data. In: Proc. of the Advances in Knowledge Discovery and Data Mining. 2009. 831−838. [doi: 10.1007/978-3-642-01307-2_86]

[17] He Z, Xu X, Huang Z, Deng S. FP-outlier frequent pattern based outlier detection. Computer Science and Information Systems, 2005,2(1):103−118. [doi: 10.2298/csis0501103h]

[18] Vreeken J, Leeuwen M, Siebes A. Krimp: Mining itemsets that compress. Data Mining and Knowledge Discovery, 2011,2(1):169−214. [doi: 10.1007/s10618-010-0202-x]

[19] Muller E, Schiffer M, Seidl T. Statistical selection of relevant subspace projections for outlier ranking. In: Proc. of the 27th IEEE Int'l Conf. on Data Engineering. Washington: IEEE, 2011. 434−445. [doi: 10.1109/icde.2011.5767916]

[20] Jiang M, Cui P, Beutel A, Faloutsos C, Yang S. Inferring strange behavior from connectivity pattern in social networks. In: Proc. of the Advances in Knowledge Discovery and Data Mining. 2014. 126−138. [doi: 10.1007/978-3-319-06608-0_11]

[21] Jiang M, Beutel A, Cui P, Hooi B, Yang S, Faloutsos C. Spotting suspicious behaviors in multimodal data: A general metric and algorithms. IEEE Trans. on Knowledge and Data Engineering, 2016,28(8):2187−2200. [doi: 10.1109/tkde.2016.2555310]

[22] Eswaran D, Gnnemann S, Faloutsos C, Makhija D, Kumar M. Zoobp: Belief propagation for heterogeneous networks. Proc. of the VLDB Endowment, 2017,10(5):625−636. [doi: 10.14778/3055540.3055554]

[23] Costa A, Yamaguchi Y, Traina A, Faloutsos C. RSC: Mining and modeling temporal activity in social media. In: Proc. of the 21st ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. Sydney: ACM, 2015. 269−278. [doi: 10.1145/2783258.2783294]

[24] Breiman L. Random forests. Machine Learning, 2001,45:5−32. [doi: 10.1007/0-387-21529-8_16]

郑皎凌(1981－),女,重庆人,博士,副教授,CCF 专业会员,主要研究领域为人工智能,数据库,知识工程.

乔少杰(1981－),男,博士后,教授,CCF 高级会员,主要研究领域为移动数据库,数据挖掘.

舒红平(1974－),男,博士,教授,博士生导师,主要研究领域为数据库,知识工程.

应广华(1988－),男,硕士,主要研究领域为人工智能,在线金融风险控制.

Louis Alberto GUTIERREZ (1980－),男,博士,Researcher,主要研究领域为数据挖掘.