基于 MS 关联规则数据挖掘模型的应用与探讨

刘城霞[1,2]
(1. 北京信息科技大学 计算机学院,北京 100101;
2. 北京邮电大学 计算机学院,北京 100876)

摘 要:文中研究了数据挖掘算法中的 MS 关联规则算法以及其在金融领域的应用。数据挖掘的作用就是要从海量的数据里找到有用的、潜在的信息,模型通过对客户账户及交易数据的过滤和深入挖掘,建立了一个为银行管理人员提供更好的智能决策和建议,为普通客户提供咨询的数据挖掘商业应用实例系统。系统的选择 Visual Studio. NET 2008 进行客户端的开发,使用 ADOMD. NET 对象连接挖掘模型和建立预测目标,使用 Web 控件对展示模型的结果。客户通过输入一些个人属性以及办理业务的基本要求,查看所关心的支付情况、贷款数量和应办理的信用卡类型,银行可以针对用户的支付特点,提供相应的增值服务等。在整个实例系统的构建过程中,对关联规则模型的挖掘过程进行了详细的分析,促进了数据挖掘的应用实践。

关键词:关联规则;数据挖掘;预测;实例系统
中图分类号:TP311 **文献标识码**: A **文章编号**:1673-629X(2013)01-0025-04
doi:10.3969/j.jssn.1673-629X.2013.01.007

Application and Discussion of Data Mining Model Based on Microsoft Association Rules Algorithm

LIU Cheng-xia[1,2]
(1. Computer School, Beijing Information and Technology University, Beijing 100101, China;
2. Computer School, Beijing University of Post and Telecommunications, Beijing 100876, China)

Abstract: The application of Microsoft association rules algorithm of data mining in financial field is discussed in this paper. The function of the data mining is mining useful and potential information from the massive data. A business data mining system is created based on Microsoft association rules algorithm, which can provide better decisions and recommendations for the bank through filtering and mining the customers´ transaction information. The client part of the system is developed with the Visual Studio. NET 2008. And it uses the objects of ADOMD. NET to associate the data warehouse and the interface and the Web controls to display the result of mining. By using the application system analyze the customer´s attributes to predict the payment ability and credit card type. The bank also can supply more service based on the customer´s interest. In the creation of the instance model system the whole program of data mining is introduced in detail and this helps the development of data mining´s application.

Key words: association rules algorithm; data mining; prediction; application system

0 引 言

数据挖掘(Data Mining)是从海量的、有噪声的干扰的、不完全的、模糊的、随机的数据中,提取隐含的、事先不知道的、而又潜在有用的信息和知识的一个过程[1]。数据挖掘是目前人工智能和数据库领域研究的热点,它可以自动地分析企业的数据,做出归纳性的推理,从中找出潜在的模式,帮助决策者调整市场策略,减少风险,做出正确的决策。在当今丰富多样的信息渠道和日益激烈的市场竞争下,企业需要留住已有的客户并且吸引新的客户。而企业取得成功的关键是需要对数据库中的海量业务数据进行数据抽取、分析转换和模型化处理后,得到帮助判断的关键性的数据,为商业决策提供真正有价值的信息,这有利于商业运作,进而获得更高的利润。很多国内外学者对金融、证卷、保险等行业信息进行数据挖掘[2,3],帮助商业管理者进行决策。

文中主要研究关于关联规则的数据挖掘模型的设计和实现,通过对银行金融数据的分析,找出有用的规

收稿日期:2012-05-22;**修回日期**:2012-08-25
基金项目:北京市人才强教计划-骨干教师资助项目(PHR201008428);北京市教委科技发展计划项目(KM201110772013);北京市优秀人才培养资助项目(2010D005007000003)
作者简介:刘城霞(1978-),女,山东泰安人,讲师,博士研究生,主要研究方向为数据挖掘、数据融合、信息安全及数据结构与算法。

则和模式,直观地反映客户信息,找出潜在的关联和预测信息,帮助银行决策人员做出正确的判断,对银行制定未来的计划提供有效的帮助。

1 关联规则算法介绍

数据关联是数据库中存在的一类重要的可被发现的知识。若两个或多个变量的取值之间存在某种规律性,就称为关联,它可分为简单关联、时序关联、因果关联。关联分析的目的是找出数据库中隐藏的关联网。近年来围绕关联规则的研究主要集中于两个方面,即扩展经典关联规则能够解决问题的范围,改善经典关联规则挖掘算法效率和规则[4,5]。

1.1 Apriori 算法

Apriori 算法[6]是 Agrawal 等人在 1993 年提出的关联规则挖掘算法。它是一种最有影响的挖掘布尔关联规则频繁项集的算法。其核心是使用候选项集找频繁项集,这里所有支持度大于最小支持度的项集称为频繁项集,简称频集。

算法的基本思想是:首先找出所有的频集,这些项集出现的频繁性至少和预定义的最小支持度一样。然后使用找到的频集产生期望的规则,产生只包含集合的项的所有规则,其中每一条规则的右部只有一项。一旦这些规则被生成,那么只有那些大于用户给定的最小可信度的规则才被留下来。为了生成所有频集,使用了递推的方法。

(1) $L1$ = find_frequent_1-itemsets(D);
(2) for ($k = 2; Lk - 1 \neq \Phi$; $k++$) {
(3) Ck = apriori_gen($Lk - 1$, min_sup);
(4) for each transaction $t \in D$ {//scan D for counts
(5) Ct = subset(Ck, t);//get the subsets of t that are candidates
(6) for each candidate $c \in Ct$
(7) $c.count++$;
(8) }
(9) $Lk = \{c \in Ck \mid c.count \geq \text{min\_sup}\}$
(10) }
(11) return $L = \cup k\ Lk$;

但 Apriori 算法也有缺点,它会产生大量的候选集,以及重复扫描数据库,这就使得很多学者研究如何去改进 Apriori 算法[7~9],或者研究其他的关联规则算法。

1.2 基于划分的算法

基于划分的算法先把数据库从逻辑上分成几个互不相交的块,每次单独考虑一个分块并对它生成所有的频集,然后把产生的频集合并,用来生成所有可能的频集,最后计算这些项集的支持度[10]。这里分块的大小选择要使得每个分块可以被放入主存,每个阶段只需被扫描一次。而算法的正确性是由每一个可能的频集至少在某一个分块中是频集保证的。该算法是可以高度并行的,可以把每一分块分别分配给某一个处理器生成频集。产生频集的每一个循环结束后,处理器之间进行通信来产生全局的候选 k-项集。通常这里的通信过程是算法执行时间的主要瓶颈;而另一方面,每个独立的处理器生成频集的时间也是一个瓶颈。

1.3 FP-树频集算法

针对 Apriori 算法的固有缺陷,J. Han 等提出了不产生候选挖掘频繁项集的方法:FP-树频集算法。它采用分而治之的策略,在经过第一遍扫描之后,把数据库中的频集压缩进一棵频繁模式树(FP-tree),同时依然保留其中的关联信息,随后再将 FP-tree 分化成一些条件库,每个库和一个长度为 1 的频集相关,然后再对这些条件库分别进行挖掘。当原始数据量很大的时候,也可以结合划分的方法,使得一个 FP-tree 可以放入主存中。基于 FP-tree 的算法的研究也十分广泛,现在对它的改进和研究仍在继续[11,12]。

1.4 Microsoft 关联规则算法

Microsoft 关联规则算法使用的是 Apriori 算法,它不分析模式,而是生成“候选项集”,然后计算该项集的数目[13]。该算法使用两个参数(support 和 probability)来说明项集以及该算法生成的规则。例如,假定 X 和 Y 表示购物车中的两个项,则 support 参数是数据集中同时包含这两个项(X 和 Y)的事例的数目。通过将 support 参数与用户定义的 MINIMUM_SUPPORT 和 MAXIMUM_SUPPORT 参数结合使用,该算法可控制生成的项集数。probability 参数(也称为“置信度”)表示数据集中既包含 X 也包含 Y 的一部分事例。通过将 probability 参数与 MINIMUM_SUPPORT 参数结合使用,可控制生成的规则数。

2 数据挖掘系统的构建

Microsoft 关联规则算法是提供参数来控制其自身使用的数据。根据要分析的数据类型,项目可表示事件、产品或属性值。最常见的关联模型类型布尔变量下,表示将 Yes/No 或 Missing/Existing 值分配给每个属性,如产品名称或事件名称。然后该算法为每个项集创建表示支持和置信度的分数。这些分数可用于排名以及从项集中获取感兴趣的规则。也可以为数值属性创建关联模型。如果属性是连续的,则可以将数值“离散化”或使用存储桶对其进行分组。而后即可将离散化值作为布尔值或属性值对来处理。

模型可能会在数据集中找到许多规则, Analysis Services 为创建的每个规则输出一个指示其“重要性”

(也称为“提升”)的分数。规则重要性的计算方法为:在已知规则左侧的情况下,求规则右侧的对数可能性值。例如,如果规则为 If {A} Then {B},则计算具有 A 和 B 的事例与具有 B 但不具有 A 的事例之比,然后使用对数刻度将该比率规范化。

2.1　系统需求分析

本系统中以银行业务数据作为挖掘基础。银行人员希望找出谁是好的客户(提供更好的服务),谁是坏的客户(减少银行的损失),进而改进银行的服务,从而给银行赚取更多的利益。对于用户而言,主要关注的是自己的相关信息、办卡情况及申请新的贷款成功的概率。银行业务数据挖掘系统功能如图 1 所示:

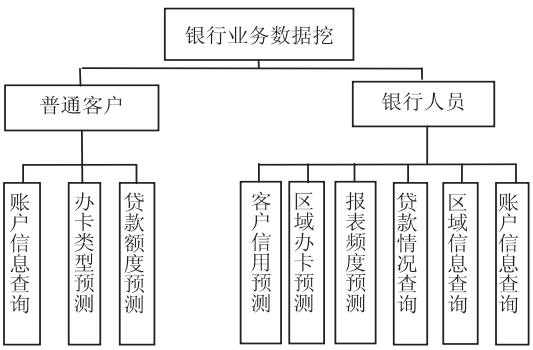

图 1　银行业务数据挖掘系统功能图

在系统设计时除了注重客户的基本需求功能外,还要尽可能的将预测的数据清晰明了的展现给用户以方便分析和制定更好的营销方案,要求:

a)预测结果是用户需要的信息。

b)提供用户完整的信息查询及操作功能。

c)允许用户从多角度查看结果。

d)系统具有可扩展性。

2.2　建立数据挖掘模型

2.2.1　数据库表介绍

数据库中包含账户(account)表,客户(client)表,支出(disposition)表,固定订单(Permanent order)表,交易(transactions)表,贷款(loan)表,信用卡(Credit card)表,地区(district)表,共八张表。数据库表间关系图如图 2 所示:

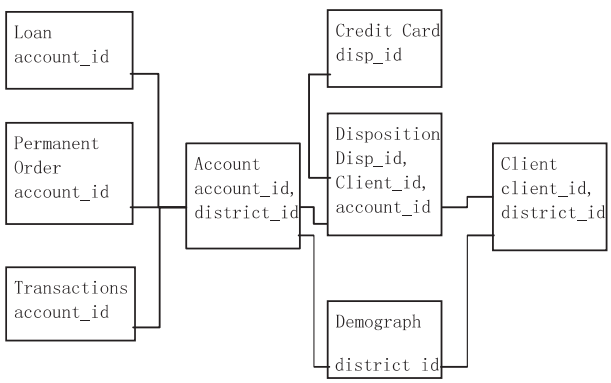

图 2　数据库表间关系图

2.2.2　建立数据挖掘预测模型

根据用户的需求,银行人员可以对客户还贷状态、客户信用、区域办卡情况、报表频度等进行预测。而客户本身可以对办卡类型、贷款额度等进行预测。

下面以银行人员预测客户还贷状态为例,建立数据挖掘预测模型。在建立模型之前必须确定用那些属性列作为输入列,输入列应该是和预测列正相关。输入列过少,根据建立去的模型得到的预测值就会和实际情况相差很多,预测的结果也就不可信。输入列过多,模型训练和预测时都会花费较多的时间,而且当数据库中的数据存在较多不包含某一输入列时也会影响预测结果,使预测结果和实际值相差很多,模型的预测结果变的不可信。所以选择合适的输入列对模型的建立是至关重要的。根据对数据库中数据的分析,和现实中的实际情况,对还款状态有影响的列主要包括客户所在地区、客户出生日期、性别、贷款的持续时间和贷款数量等。下面用于对还款状态进行预测的模型就是以地区、出生日期、性别、贷款持续时间、贷款数量作为输入列的。DMX 建模语言如下:

```
create mining modelloan_status_evaluate(
account_id text key,          //账户 ID
district_id text discrete,    //客户所在地区
birth_number text discrete, //出生日期
duration text discrete,       //贷款期限
amount text discrete ,        //贷款数量
[status] text discrete predict //贷款状态
)
usingMicrosoft_Association_Rules
```

建立起模型后,要对模型进行评估,为此,将数据库中的数据分成两部分,其中一部分为训练数据,用于对模型进行训练,另外一部分是预测数据,用于对模型进行评估。通过对预测数据的预测,将得到的预测值和实际值进行比较,就可以得到模型质量的好坏。

将 account 表中的数据分为两部分,分别放到 account_evaluate 表中,用于模型的预测,和 account_exercise 表中,用于模型的训练。其他的和输入列有关的数据也都分成两部分。

首先用 account_exercise 表中的数据对该挖掘模型进行训练。训练语句如下:

```
Insert into loan_status_evaluate(account_id,district_id,birth_number,duration,amount,[status])
openquery([Warehouse],
'select account_exercise.account_id, account_exercise.district_id,
client.birth_number,
loan.duration,
```

```
loan. amount,
loan. status
from dbo. account_exercise,client,loan,disp
where account_exercise. account_id = loan. account_
id
and account_exercise. account_id = disp. account_id
and client. client_id = disp. client_id ′)
```

训练完成后用该挖掘模型对进行预测,在预测结果中得到可能的还款状态和相对应该还款状态出现的可能性概率。DMX 的预测语句如下:

```
Select t. account_id,t. district_id,t. birth_number,
t. duration,t. amount,predict([status]),
predictprobability([status])
from loan_status_evaluate prediction join
open query([Warehouse],
′Select account_evaluate. account_id, account_evalu-
ate. district_id,client. birth_number,loan. duration,loan.
amount,loan. status
from dbo. account_evaluate,client,loan,disp
where account_evaluate. account_id = loan. account_
id
and account_evaluate. account_id = disp. account_id
and client. client_id = disp. client_id ′) as t
on loan_status_evaluate. account_id = t. account_id
and loan_status_evaluate. district_id = t. district_id
and loan_status_evaluate. birth_num= t. birth_num
and loan_status_evaluate. duration = t. duration
and loan_status_evaluate. amount = t. amount
and loan_status_evaluate.[status] = t.[status]
```

评估结果如图 3 所示:

消息 结果

account_id	district_id	birth_numl	duration	amourt	Expression	Expression
8519	23	"500115"	12	74688	"A"	0.8557377...
8523	54	"805912"	12	93036	"A"	0.8557377...
8533	47	"525216"	48	174048	"C"	1
8547	14	"455223"	36	173016	"C"	1
8558	1	"351215"	60	288360	"C"	0.7366782...
8564	68	"490107"	24	76680	"C"	0.9274509...
8564	68	"545319"	24	76680	"C"	0.9274509...
8566	74	"426109"	36	230220	"C"	1
854	62	"545316"	48	87216	"C"	1

图 3 还贷状态预测模型评估结果

由结果可以看出,建立的预测模型基本合理,通过训练集的训练和测试集的测试评估,结果基本符合要求,能够正确的预测贷款状态。

3 结果分析

系统在网页上用友好界面展示了挖掘结果。

图 4 为预测还款状态的挖掘模型 loan_status 所挖掘出的频繁项集,是挖掘出的所有满足一定支持度的频繁项集的部分结果。例如:对于图中所示第一行数据,表明贷款数量(amount)为 89340,客户所在地编号(district_id)为 60,贷款持续时间(duration)为 60,就是满足所要求支持度的频繁项集。

挖掘出的所有频繁项集
amount = 89340, district_id = 60, duration = 60
amount = 89340, district_id = 60, status = "C"
amount = 89340, duration = 60, status = "C"
amount = 428784, district_id = 1
amount = 428784, duration = 48
amount = 428784, status = "C"

图 4 挖掘出的频繁项集

另外,通过这个预测还款状态的挖掘模型所挖掘出的规则如图 5 所示:

挖掘出的所有规则
amount = 88704, district_id = 23 -> status = "C"
amount = 66840, district_id = 15 -> status = "C"
amount = 66840, duration = 24 -> status = "C"
amount = 130896 -> status = "C"
amount = 130896, district_id = 38 -> status = "C"

图 5 挖掘出的规则

对于图中第一行数据,其含义为:在贷款数量(amount)为 88704,客户所在地区编号(district_id)为 23 时最有可能的还款状态(status)是“C”。

图 6 中左侧数据表是预测还款状态的挖掘模型的预测结果,比如:账号为 330 的账户,最可能的还款状态是“C”,而且还款状态为“C”的概率为 0.8541。右侧的数据是账户的实际还款状态,账号为 330 的账户的实际还款状态为“C”。当然预测模型不是百分之百正确,其中也有预测出错的情况,比如第一行数据,预测还款状态为 A,实际还款状态为 B。

账号	预测的还款状态	可能性	账号	实际
3273	"A"	0.655737704918033	3273	"B"
3293	"C"	0.636678200692042	3293	"C"
330	"C"	0.851166666666667	330	"C"
330	"C"	0.851166666666667	330	"C"
3329	"A"	0.655737704918033	3329	"A"

图 6 预测还款状态和实际还款状态

通过对预测的还款状态和实际的还款状态的比较可以很容易的确定模型质量的好坏。如果通过挖掘模型预测出的数据和实际的数据相差无几,即可以认为该挖掘模型是良好的,挖掘出的知识是可信的。相反,如果通过挖掘模型预测的数据和实际的数据相差很多,那么该模型就是不可取的,必须更改输入列、重新建立模型,重新对模型评估。

系统还有其他的预测功能,篇幅关系不再赘述。

(下转第 33 页)

评分放在 Web 服务组内完成,并区分了影响服务质量的客观因素和主观因素,并根据客观影响因素,提出了 Web 服务组内原子服务的动态 QoS 计算方法。实验证明,组内 QoS 动态计算方法能够实时有效地获得组内原子服务的 QoS,避免了不同功能组内原子服务状态的变化对其他功能组的影响。下一步,将尝试把主观影响因素整合到现有的动态 QoS 计算公式内,进一步完善 QoS 的度量方法,建立服务质量的度量指标体系以及服务组合质量的动态保障机制。

参考文献:

[1] Web Services Activity[EB/OL]. 2002. http://www.w3.org/2002/ws.

[2] WSDL. Web Services Description Language. 1.1. [EB/OL]. 2001-03. http://www.w3.org/TR/wsdl.

[3] W3C. SOAP Specification[EB/OL]. 2007. http://www.w3.org/TR/soap/.

[4] 岳　昆,王晓玲,周傲英. Web 服务核心支撑技术:研究综述[J]. 软件学报,2004,15(3):428-442.

[5] 杨胜文,史美林. 一种支持 QoS 约束的 Web 服务发现模型[J]. 计算机学报,2005,28(4):589-594.

[6] Comuzzi M,Pernici B. A framework for QoS-based Web service contracting[J]. ACM Trans. on Web,2009,3(3):1252-1254.

[7] Lin Chia-Feng,Sheu Ruey-Kai,Chang Yue-Shan,et al. A relaxable service selection algorithm for QoS-based web service composition[J]. ELSEVIER Information and Software Technology,2011,53(12):1370-1381.

[8] Stephen S,Yau Yinyin. QoS-based Service Ranking and Selection for Service-based Systems[C]//2011 IEEE International Conference on Services Computing. [s.l.]:[s.n.],2011.

[9] Yang S J H,Hsieh J S F,Lan B C W,et al. Composition and Evaluation of Trustworthy Web Services[J]. International Journal of Web and Grid Services,2006,2(1):5-24.

[10] Vambenepe W,Thompson C,Talwar V,et al. Dealing with Scale and Adaptation of Global Web Services Management [C]//Proceedings of the IEEE International Conference on Web Services(ICWS'05). Orlando:[s.n.],2005:339-346.

[11] Ran S. A Model for Web Services Discovery with QoS[J]. ACM SIGEcom Exchanges,2003,1(4):1-10.

[12] Wang Xia,Vitvar T,Kerrigan M,et al. A QoS-aware Selection Model for Semantic Web Services[C]//Proceedings of the 4th International Conference on Service-oriented Computing. [s.l.]:[s.n.],2006:390-401.

[13] 邵凌霜,周　立,赵俊峰,等. 一种 Web Service 的服务质量预测方法[J]. 软件学报,2009,20(8):2062-2073.

[14] 黄景文,胡志华. Web 服务 QoS 的免疫多信号预测模型研究[J]. 广西大学学报(自然科学版),2009,34(4):535-539.

[15] Papazoglou M P,Paolo T,Schahram D,et al. Service-oriented Computing:State of the Art and Research Challenges[J]. IEEE Computer,2007,40(11):38-45.

(上接第 28 页)

4　结束语

人们对数据的应用从简单的数据查询,到从海量数据中挖掘知识并提供关联的服务,这就是数据挖掘的作用。文中针对基于关联算法的数据挖掘系统进行了分析和设计,并实现了一个简单的但是完整的银行业务数据挖掘系统,对客户账户情况、贷款情况等进行信息的预测,促进了基于关联的数据挖掘系统的应用研究。

参考文献:

[1] 安淑芝. 数据仓库与数据挖掘[M]. 北京:清华大学出版社,2005.

[2] 赵裕啸,倪志伟,王园园,等. SQL Server 2005 数据挖掘技术在证卷客户忠诚度的应用[J]. 计算机技术与发展,2010,20(2):229-232.

[3] 陈　艳,张燕平. 数据挖掘技术在保险客户信用评估的应用[J]. 计算机技术与发展,2008,18(5):179-181.

[4] 宋宝莉,覃　征. 分布式全局频繁项目集的快速挖掘方法[J]. 西安交通大学学报,2006,40(8):923-927.

[5] 唐　瑜,王　勇. 挖掘最大频繁项集的优化方法[J]. 计算机工程与应用,2006,42(31):171-173.

[6] 邵峰晶,于忠清. 数据挖掘原理与算法[M]. 北京:中国水利水电出版社,2003:123-135.

[7] 黄　进,尹治本. 关联规则挖掘的 Apriori 算法的改进[J]. 电子科技大学学报,2003,32(1):76-79.

[8] 朱孝宇,王理东,汪光阳. 一种改进的 Apriori 挖掘关联规则算法[J]. 计算机技术与发展,2006,16(12):89-90.

[9] 李绪成,王保保. 挖掘关联规则中的 Apriori 算法的一种改进[J]. 计算机工程,2002,28(7):104-105.

[10] Han Jiawei,Kamber M. 数据挖掘概念与技术[M]. 范　明,孟小峰译. 北京:北京机械工业出版社,2001:149-180.

[11] 宋余庆,朱玉全,孙志挥,等. 基于 FP-tree 的最大频繁项目集挖掘及更新算法[J]. 软件学报,2003,14(9):1586-1592.

[12] 凌绪雄,王社国,李　洋,等. 无项头表的 FP-Growth 算法[J]. 计算机应用,2011,31(5):1391-1394.

[13] Microsoft 关联算法技术参考[EB/OL]. 2008. http://msdn.microsoft.com/zh-cn/library/cc280428.aspx.