基于 Apriori 改进算法的企业 Web 日志挖掘研究

吴红星,王 浩
(合肥工业大学 计算机与信息学院,安徽 合肥 230009)

摘 要:由于企业的 Web 日志中隐藏着大量有价值的信息,Apriori 算法的缺点在于产生大量的候选集以及频繁扫描数据集,文中是基于协同门户和网站的日志信息进行研究。企业的协同门户里企业通知栏目可以随时发布企业的相关通知信息,是企业第一时间想让用户看到的。而网站里企业的新闻栏目也是想给用户展示企业的相关新闻信息和企业的经营活动信息,完成企业品牌以及企业文化的宣传等。基于协同门户和网站在企业的这点共性,文中提出了针对企业的一种改进 Apriori 算法,即在企业主动向访问者展现通知公告或者企业的经营新闻信息的前提下,挖掘出其他一级主栏目在访客心中的地位,以及访客对这些栏目的关注度和兴趣度,以便于企业实现如何调整其他栏目布局,更好地为企业宣传做服务,同时又能满足访问者的便捷访问,等等。文中算法改进的核心思想是减少候选集来对 Apriori 算法进行改进。在 Apriori 算法的扫描过程中,某个 ID 不参与,当算法挖掘出最大频繁集后再将这个 ID 添加到最大频繁项集的每个项集中,开展关联规则的挖掘。这样在数据集的扫描次数及候选集的产生上都有较大程度的优化。对比实验结果表明,改进的 Apriori 算法效果明显,对企业有较强的实际应用价值。

关键词:Web 应用;日志;关联规则;算法改进;Apriori 算法
中图分类号:TP301.6 **文献标识码**:A **文章编号**:1673-629X(2015)04-0043-05
doi:10.3969/j.issn.1673-629X.2015.04.011

Research on Enterprise Web Log Mining Based on Improved Apriori Algorithm

WU Hong-xing, WANG Hao
(School of Computer and Information, Hefei University of Technology, Hefei 230009, China)

Abstract: A large number of valuable information is hidden in the enterprise Web log, the disadvantage of Apriori algorithm is to produce a large number of candidate set and frequent scan data set. In this paper, study based on Web log information from collaborative Web portal. The enterprises collaborative Web portal can release the relevant notice of enterprise information at the announcements column at any time, which is what the enterprise want visitors to see at the first time. The Website news is to show visitors for enterprise related news, information and enterprise management activities, it's also to complete the enterprise brand and enterprise culture propaganda, etc. Based on the general character of collaborative Web portal, present an improved Apriori algorithm for enterprises, the enterprises show visitors announcements or business news and information actively, dig out the status of the other main column in visitors, and the degree of these columns' attention and interest in visitors. In this way, the enterprises can adjust the other column layout, do better service for enterprise propaganda, and meet the visitors' convenient access, etc. The core of the improved algorithm is to reduce the candidate set. In the process of scanning of Apriori algorithm, an ID is not to participate in, when the algorithm mining the maximum frequent sets and then adding the ID to the maximum frequent item sets concentration of each item, to carry out the association rules mining. There is a larger degree of optimization in the number of data sets of scanning and candidate set generation. After the contrast experiments, it shows that the improved Apriori algorithm is effective and has the strong practical application value for enterprises.

Key words: Web applications; log; association rules; improved algorithm; Apriori algorithm

收稿日期:2014-06-19 **修回日期**:2014-09-25 **网络出版时间**:2015-02-23
基金项目:国家"973"重点基础研究发展计划项目(2013CB329604);国家"863"高技术发展计划项目(2012AA011005)
作者简介:吴红星(1974-),男,博士研究生,研究方向为信息系统集成、数据库、数据挖掘;王 浩,博士,教授,研究方向为信息系统、数据挖掘。
网络出版地址:http://www.cnki.net/kcms/detail/61.1450.TP.20150223.1241.039.html

0 引 言

当今关于大数据、云的研究愈发深入,可是如何用好这些数据,发现这些数据背后隐藏的信息却成为更具实际价值的工作,这也就是数据挖掘的概念。简单说,数据挖掘就是通过分析大量的数据来揭示有意义的某种关系、趋势和模式的过程,数据挖掘的整个过程就是从大量的、不完全的、有噪声的、模糊的、随机的数据中,提取隐藏在其中而又事先不知道的,但又是潜在有用的信息和知识。数据挖掘是决策支持过程,是一门交叉性学科,涉及人工智能、数据库技术、机器学习、知识发现、统计学、可视化技术等等[1]。其与数据分析最大的区别在于未知性,当前数据挖掘已经在社会的各行各业深度应用。“9·11”事件后,美国启动了“全面信息感知(TIA)”计划,它将通过各种公开信息的搜集,包括所有人的上网行为、信用卡记录、健康档案、学习成绩、出行时间等等,加以分析,获取有价值的情报。这也是人类有史以来最大规模的信息监控系统以及数据挖掘系统[2]。斯诺登事件更可以看出数据挖掘已经应用到国家层面的信息战。

数据挖掘的任务就是发现隐藏在数据中的模式。可分为分类模式、聚类模式、回归模式、关联模式、序列模式、偏差模式等[3]。Web 挖掘可以分为内容、结构、使用记录等[4]。文中从企业实际应用出发,以企业门户为研究对象,本着门户是企业的展示窗口,是访问者获知企业基本情况、经营情况、企业文化等的主要公共官方途径。为了让企业门户的展现更加符合企业开设门户的初衷,同时也能符合访问者的浏览喜好,结合企业的门户日志,进行数据挖掘。通过对 Web 日志数据的清洗及过滤,得出日志最核心的信息,并结合企业门户的实际情况提出改进的 Apriori 算法,挖掘其中某些栏目之间可能存在的直接或间接关系,从而更加快速、有效地帮助企业分析访问者的兴趣度,建立关联规则模型,从而优化门户版面结构,提高门户满意度。

1 Apriori 算法

Apriori 算法作为经典的关联规则挖掘算法,其核心内容为连接与剪枝操作。该算法是 Agrawal 和 R. Srikant 于 1994 年提出的,为布尔关联规则挖掘频繁项集的原创型算法[1]。算法的核心是用前一次扫描数据库的结果产生本次扫描的候选项目集,再通过逐次扫描和迭代最终得到最大频繁项集[5-6]。一旦数据库中的事务找出频繁项集,就可以直接由它们产生强关联规则,这些强关联规则满足最小支持度和最小置信度[1]。所以 Apriori 算法涉及到支持度(Support)和置信度(Confidence)这两个概念。

支持度:$\mathrm{Support}(A \rightarrow B) = P(A \cup B)$,表示 A 与 B 同时出现的概率。如果 A 与 B 同时出现的概率小,说明 A 与 B 的关系不大;如果 A 与 B 同时出现的非常频繁,则说明 A 与 B 总是相关的。

置信度:$\mathrm{Confidence}(A \rightarrow B) = P(A \mid B)$,表示 A 出现时,B 是否也会出现或有多大概率出现。如果置信度为 100%,则说明只要 A 出现 B 则一定出现,如果置信度太低,则说明 A 的出现与 B 是否出现关系不大。

Apriori 算法[1]使用一种称为逐层搜索的迭代方法,其中 K 项集用于探索 $(K+1)$ 项集。首先,通过每次扫描数据库,统计每个项的计数,收集满足最小支持度的项,找出频繁 1 项集的集合,然后再使用频繁 1 项集的集合找出频繁 2 项集的集合。以此类推,直到不能再找出频繁项集。

Apriori 算法具体的执行步骤如下:

(1)首次扫描数据集生成候选集 C_1,通过逐层扫描统计候选集中每个项集 X 的支持度,删除 $X.\,\mathrm{support} < \mathrm{minsupport}$ 的项集得到频繁集 L_1;

(2)频繁集 L_1 再进行自身连接生成候选集 C_2,再次通过逐层扫描,删除 $X.\,\mathrm{support} < \mathrm{minsupport}$ 的项集得到频繁集 L_2;

(3)对 $k \geqslant 2$ 的每个候选集 C_k,重复第 2 步,最终得出最大频繁项集 L_k。

从以上的步骤可以看出,找出每个频繁项集的时候都需要完整地扫描一次数据库,效率非常低下。

如何能减少 Apriori 算法扫描次数及候选集的数量,是算法改进的关键问题[7-9],也是提高 Apriori 算法效率的关键。针对怎样才能进一步提高基于 Apriori 算法的挖掘效率,已经有如下的技术和方法:基于散列的技术、事务压缩方法、划分技术、抽样方法、动态项集计数技术等等[1]。目前,有很多对 Apriori 算法改进的文献研究,被用于日志关联规则的发现:易芝等对基于关联规则相关性分析的 Web 个性化推荐做了研究[10];孙赵平等对 Apriori 算法的项目集进行了缩减,提高了挖掘效率和稳定性[11];孟庆川和陈晓明利用云理论对日志挖掘中的 Apriori 算法进行了改进[12];习慧丹对动态 Web 日志挖掘问题做了分析与研究[13];曹莹等运用精减频繁项集、多关键字排序重排频繁项集等方法对 Apriori 算法进行改进[14]。

2 改进的 Apriori 算法

Apriori 算法的缺点在于产生大量的候选集以及频繁扫描数据集,文中是基于协同门户和网站的日志信息进行研究。协同门户里企业通知栏目可以随时发布企业的相关通知信息,是企业第一时间想让用户看到的。企业的网站里企业的新闻栏目也是想给用户展示集团及所属企业的相关新闻信息及企业的经营活动信

息,完成企业品牌以及企业文化的宣传等。企业门户的首页已经将所有的一级栏目都设置了相应独立的访问入口,正是由于具备这样的条件,通过简单的 SQL 统计发现企业的通知公告和企业的新闻栏目是访问量最多的栏目。基于协同和网站在企业的这点的共性,文中提出了针对企业的一种改进 Apriori 算法,即在企业主动向访问者展现通知公告或者企业的经营新闻信息的前提下,挖掘出其他一级主栏目在访客心中的地位,以及访客对这些栏目的关注度和兴趣度。以便于以企业通知公告或者新闻为主展示窗口,在访问了通知公告或者企业新闻栏目的同时最关注和访问了其他哪些栏目,最终实现如何调整其他栏目布局,更好地为企业宣传做服务,同时又能满足访问者的便捷访问,等等。文中算法改进的核心思想是减少候选集来对 Apriori 算法进行改进。

综上所述,文中以企业的网站日志数据为实验对象。结合现状对网站日志数据进行相关挖掘工作,将栏目 ID 作为属性挖掘栏目之间的关联性,主要挖掘其他栏目与企业新闻栏目(设其 ID 为 A)的关联性,所以选取日志中均包含访问 A 的日志记录,在 Apriori 算法的扫描过程中,A 的 ID 不参与,当算法挖掘出最大频繁集后再将 A 的 ID 添加到最大频繁项集的每个项集中,开展关联规则的挖掘。这样在数据集的扫描次数及候选集的产生上都有较大程度的优化。

Apriori 算法改进后的流程图如图 1 所示。

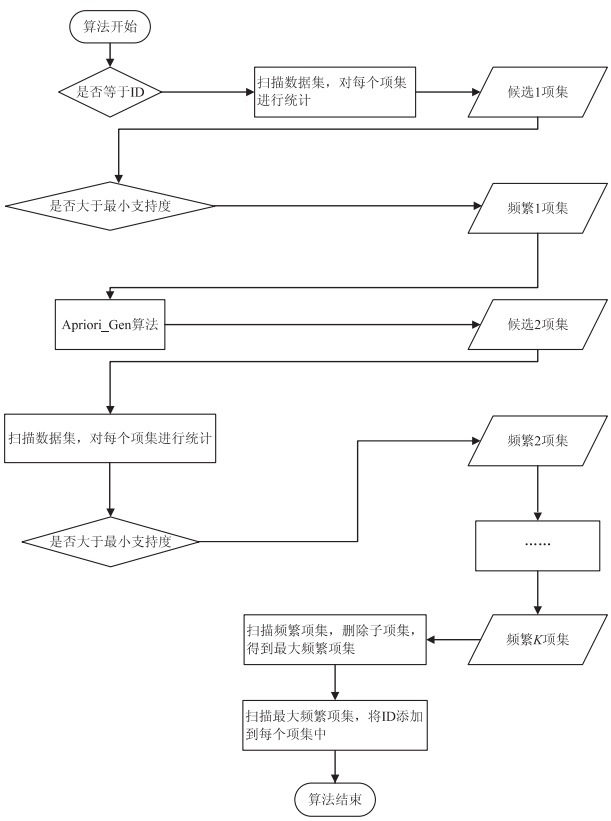

图 1　改进的 Apriori 算法的流程图

Apriori 改进算法的伪代码描述如下:

(1) (C_1 = {candidate$_1$ – itemsets | itemsets!="id"});

(2) $L_1 = \{c \in C_1 \mid c.\text{count} \geqslant \text{minsupport}\}$;

(3) for($k = 2, L_{k-1} \neq \varnothing, k++$) //直到不能再生成最大项目集为止

(4) C_k = apriori_gen(L_{k-1});// 生成含 k 个元素的侯选项目集

(5) for all transactions $t \in D$ do begin //开始处理

(6) C_t = count_support($C_{k,t}$);// 包含在事务 t 中的侯选项目集

(7) for all candidates $c \in C_t$

(8) c. count++;

(9) end

(10) $L_k = \{c \in C_k \mid c.\text{count} \geqslant \text{minsupport}\}$;

(11) end

(13) key = key ∪ L_k

(14) for all transactions $x \in$ key do begin

(15) key = key + "id"

(16) end

3　改进的 Apriori 算法实例

3.1　数据的准备

在企业网站的 Web 日志选取最近半年内共一千多万条日志记录,利用 SQL Server 数据库建库,然后建表并将日志记录原样导入表中。

对这一千多万条日志数据进行清洗,将需要的关键信息提取到另外一张单独的表(表中新增一级栏目及二级栏目 ID 字段),这里面分三种情况:

(1) uriquery 字段中有新闻 ID 的,需要将 ID 提取出来,然后结合网站数据库将新闻对应的一级栏目和二级栏目 ID 赋值到相应字段;

(2) uriquery 字段中有栏目 ID 的可直接将栏目 ID 提取出来直接赋值到相应字段;

(3) 通过其他独立入口访问的记录,需要依据 uristem 的值分别手动赋值到相应字段,另外还有相当一部分的残缺、无效的脏数据因无法提取有效信息的需要手动过滤。

经过上述的数据处理过程,将日志记录清洗成所需要的数据,共计两百多万条日志记录,记录内容如图 2 所示。

3.2　关联规则模型建立

通过 MyEclipse 平台,用 Java 实现 Apriori 算法。为了简化程序运行及输出结果,将企业新闻、企业概况、企业文化、企业党建、携手企业等栏目名分别重命名为 a、b、c、d、e,原算法数据集搜索频繁集见图 3。

Pid	date	time	menid	ip	part
1	2013-12-30	01:22:20	107003	61.48.107.234	107
2	2013-12-30	01:22:20	107003	61.48.107.234	107
3	2013-12-30	01:22:20	107003	61.48.107.234	107
4	2013-12-30	01:22:21	107003	61.48.107.234	107
5	2013-12-30	01:22:21	107003	61.48.107.234	107
6	2013-12-30	01:22:21	107003	61.48.107.234	107
7	2013-12-30	01:22:21	107003	61.48.107.234	107
8	2013-12-30	01:22:22	107003	61.48.107.234	107
9	2013-12-30	01:22:22	107003	61.48.107.234	107
10	2013-12-30	01:22:22	107003	61.48.107.234	107
11	2013-12-30	01:22:23	107003	61.48.107.234	107
12	2013-12-30	01:22:23	107003	61.48.107.234	107
13	2013-12-30	01:22:23	107003	61.48.107.234	107

图 2 清洗后的日志数据

TID	Support
a	100
b	34
c	19
d	11
e	22

C_1

TID	Support
a	100
b	34
c	19
d	11
e	22

L_1

(a) 候选集C_1和频繁集L_1

TID	Support
ab	34
ac	19
ad	11
ae	22
bc	10.6
bd	6.4
be	15.6
cd	7
ce	9.2
de	5.8

C_2

TID	Support
ab	34
ac	19
ad	11
ae	22
bc	10.6
be	15.6

L_2

(b) 候选集C_2和频繁集L_2

TID	Support
abc	10.6
abd	6.4
abe	15.6
acd	6.9
ace	9.2
ade	5.8
bcd	4.7
bce	7.5

C_3

TID	Support
abc	10.6
abe	15.6

L_3

(d) 候选集C_3和频繁集L_3

图 3 候选集和频繁集

改进算法数据集搜索频繁集如图 4 所示。

TID	Support
b	34
c	19
d	11
e	22

C_1

TID	Support
b	34
c	19
d	11
e	22

L_1

(a) 改进算法后的候选集C_1和频繁集L_1

TID	Support
bc	10.6
bd	6.4
be	15.6
cd	7
ce	9.2
de	5.8

C_2

TID	Support
bc	10.6
be	15.6

L_2

(b) 改进算法后的候选集C_2和频繁集L_2

图 4 改进算法后的候选集和频繁集

设定最小支持度和最小置信度为 0.1 和 0.6 以及 0.06 和 0.6,优化后的部分程序如下:

```
public void rulePrint(){
String x,y, temp;
Set<String> hs=ruleMap.keySet();
Iterator<String> iterator=hs.iterator();
StringBuffer sb=new StringBuffer();
System.out.println("挖掘关联规则如下:");
while(iterator.hasNext()){
x=(String)iterator.next();
y=(String)ruleMap.get(x);
temp=formater.format(count_sup(x+y)/count_sup(x));
System.out.println(x+"==>"+y+"\t"+temp);
sb.append(x+(x.length()<5?"  :"")+"==>"+y+"\t"+temp+"\t"+"\n");
}
BufferedWriter bw=null;
try {
FileWriter fw=new FileWriter("Result.txt");
bw=new BufferedWriter(fw);
bw.write("开始时间:"+df.format(new Date()));
bw.newLine();
bw.write("最小支持度 minsup="+minsup);
bw.newLine();
bw.write("最小置信度 minconf="+minconf);
bw.newLine();
bw.write("挖掘的规则如下:");
bw.newLine();
bw.write(sb.toString());
bw.write("结束时间:"+df.format(new Date()));
if(bw!=null)
bw.close();}
catch (Exception e){
e.printStackTrace();}
}
……
public static void main(String[] args){
System.out.println("开始时间:"+df.format(new Date()));//new Date()为获取当前系统时间
MyApriori ap=new MyApriori(create(),0.06,0.6);
ap.run();
System.out.println("结束时间:"+df.format(new Date()));}
```

从程序运行的结果看,最小支持度和最小置信度为 0.1 和 0.6 以及 0.06 和 0.6 时,原始算法和优化后的耗时分别为:157 s,141 s 以及 156 s,141 s。可见效果明显。

4 改进的 Apriori 算法实验结果分析

4.1 实验结果分析

按照文中第 2 节的算法优化思路,对 Apriori 算法

源码进行改动,在扫描数据集时,发现固定的栏目时直接过滤,该栏目 ID 将不参与候选集与频繁集的产生,当最大频繁集产生后再将固定栏目的 ID 添加到最大频繁项集中进行关联规则的挖掘,建立关联模型,程序运行的耗时结果如表 1 所示。

表 1　算法耗时比较

	原始算法 /ms	文中方案 /ms	结果
最小支持度和最小置信度分别为 10% 和 60%	157	141	优
最小支持度和最小置信度分别为 6% 和 60%	156	141	优

观察程序运行结果,发现候选集明显减少,从原算法的最多 10 个减少为 6 个,当最小支持度和最小置信度分别为 10% 和 60% 时,改进前和改进后的运行时间分别为 157 ms 和 141 ms。当最小支持度和最小置信度分别为 6% 和 60% 时,改进前和改进后的运行时间分别为 156 ms 和 141 ms。可见改进的 Apriori 算法的优化还是比较明显的。

4.2　Web 日志挖掘的结果分析

通过分析算法挖掘建立的模型,可以看出 b(企业概况)与 a(企业新闻)之间的关联性最强,其余栏目按照同 news 关联性由强到弱依次的顺序是 e(携手企业)、c(企业文化)、d(企业党建)。这给网站栏目的布局提出一定的建议:网站栏目在保留原有 a(企业新闻)第一的前提下,将 b(企业概况)和 e(携手企业)栏目次序前移至相邻位置,d(企业党建)栏目位置放在最后,这样对于访问者的阅读习惯是合适的。同时根据挖掘的企业也要加强企业文化自身的建设,这也是访问者关注和在意的热点。

5　结束语

文中结合企业实际应用对 Apriori 算法应用提出一种改进思路,并用企业网站产生的日志数据进行对比实验,证明了优化的有效性,并通过建立关联模型,分析各栏目之间的兴趣度,对网站的布局结构提出建议,对于协同门户和网站有一定的实际指导意义。

参考文献:

[1] Han Jiawei, Kamber M, Jian Pei. 数据挖掘:概念与技术[M]. 范　明,孟小峰,译. 北京:机械工业出版社,2012.

[2] 王飞跃. 开源情报与网络时代的国家安全[EB/OL]. 2007. http://www.libnet.sh.cn/tsgxh/hyzq/list.asp? id=2898.

[3] 王光宏,蒋　平. 数据挖掘综述[J]. 同济大学学报:自然科学版,2004,32(2):246-252.

[4] 韩家炜,孟小峰,王　静,等. Web 挖掘研究[J]. 计算机研究与发展,2001,38(4):405-414.

[5] Chen Ming-Syan, Park J S, Yu P S. Efficient data mining for path traversal patterns[J]. IEEE Transactions on Knowledge and Data Engineering,1998,10(2):209-221.

[6] 刘　兵. Web 数据挖掘[M]. 北京:清华大学出版社,2009.

[7] 阳小华,周龙镶. 基于用户访问模式的 WWW 浏览路径优化[J]. 软件学报,2001,12(6):846-850.

[8] Mobasher B, Dai Honghua, Luo Tao, et al. Integrating Web usage and content mining for more effective personalization[J]. Lecture Notes in Computer Science,2000,1875:165-176.

[9] Macclennan T Z H. 数据挖掘原理与应用[M]. 北京:清华大学出版社,2007:99-107.

[10] 易　芝,汪林林,王　练. 基于关联规则相关性分析的 Web 个性化推荐研究[J]. 重庆邮电大学学报:自然科学版,2007,19(2):234-237.

[11] 孙赵平,李龙澍. 基于关联规则的 Web 日志挖掘算法研究[J]. 电子技术,2010,47(8):11-13.

[12] 孟庆川,陈晓明. 基于关联规则 Web 日志挖掘算法的研究[J]. 信息技术,2010(3):96-98.

[13] 习慧丹. Web 日志挖掘探索[C]//第三届全国软件测试会议论文与移动计算、栅格、智能化高级论坛论文集. 出版地不详:出版者不详,2009:184-186.

[14] 曹　莹,苗志刚,张红霞. 基于改进的 Apriori 算法的学位预警应用研究[J]. 电脑开发与应用,2014,27(6):1-3.

(上接第 42 页)

sium on information theory. Los Alamitis:IEEE Computer Society,2002.

[9] Bouquet P, Euzenat J, Franconi E, et al. Specification of a common framework for characterizing alignment[EB/OL]. 2004. http://www.aifb.uni-karlsruhe.de/WBS/phi/kweb-221.pdf.

[10] 徐　猛,刘宗田,周　文. 一种基于知网语义相似度计算的应用研究[J]. 微计算机信息,2010,26(1-3):200-201.

[11] 曹泽文,钱　杰,张维明,等. 一种综合的概念相似度计算方法[J]. 计算机科学,2007,34(3):174-175.

[12] 谷志锋,刘　勇,郭跟成. 基于相似度计算的本体映射优化方法[J]. 计算机工程,2008,34(19):56-57.

[13] 徐　茜,彭进业,李　展. 本体映射中一种综合的概念相似度计算方法[J]. 计算机工程与应用,2010,46(24):34-36.

[14] 马　驰,王念滨,张海燕. 基于相似度计算的本体映射框架[J]. 计算机工程,2009,35(12):61-63.

[15] Cai N, Yeung R W. A security condition for multi-source linear network coding[C]//Proceedings of 2007 IEEE international symposium on information theory. Nice:IEEE Computer Society,2007:561-565.

基于Apriori改进算法的企业Web日志挖掘研究

作者： 吴红星， 王浩， WU Hong-xing， WANG Hao
作者单位： 合肥工业大学 计算机与信息学院,安徽 合肥,230009
刊名： 计算机技术与发展 ISTIC
英文刊名： Computer Technology and Development
年，卷(期)： 2015(4)