改进 FP-growth 算法在气象预报中的应用①

刘 娟, 宋安军

(上海海事大学 信息工程学院, 上海 201306)

摘 要: 针对现在全球极端天气频发的现状, 天气预报用来及时发现灾害天气的出现显得尤为重要. 随着数据挖掘技术的迅速发展和广泛应用, 采用了改进 FP-growth 算法挖掘出各种气象因子之间可能存在的关联, 从而发现气象特点, 对近期天气气象做出预报. 经过仿真实验验证, 改进后的算法在天气预报准确率有了明显的提高.

关键词: 天气预报; 气象因子; 数据挖掘; 关联规则; FP-growth 算法

Application of an Improve FP-growth Algorithm in Meteorological Forecast

LIU Juan, SONG An-Jun

(College of Information Engineering, Shanghai Maritime University, Shanghai 201306, China)

Abstract: Aiming at the problem of the global frequent extreme weather conditions, it is more important to discover the appearance of the disaster weather. With the rapid development and wide application of data mining technology. In this paper, it uses improved FP-growth algorithm to mining the possible correlation between various meteorological factors, so as to find the weather characteristics and forecast the near future weather condition. The simulation results show that the improved algorithm has a significant improvement in the accuracy of weather forecast.

Key words: weather forecast; meteorological factor; data mining; association rules; FP-growth algorithm

1 导言

随着全球天气的不断变化, 高温干旱、暴雨等灾害天气的频发. 人们对于气象预报的准确性提出了更高的要求. 大数据时代的来临, 将数据挖掘技术应用于气象预报中, 分析各种气象因子之间的关联, 提高气象预报的准确性, 具有十分重要的现实意义.

传统的气象预报是基于统计的预测模型, 采用概率中的相关方法将历史数据建立一个或多个模型. 但是, 传统的统计方法往往适用于大量预报对象, 预报对象越多, 找出的预报因子和预报对象之间的关联越多, 得到的统计结果更加精确. 然而, 在实际应用中, 往往需要针对某一特定天气对象进行预报, 传统的统计方法存在一定的局限性. 使用数据挖掘技术可以针对某一特定天气对象, 快速处理海量天气数据, 挖掘出潜在的, 人们不易发觉的预报因子之间的关联, 大大提高了天气预报的准确性.

数据挖掘技术就是从海量数据中挖掘出人们感兴趣的、有价值的、未知的信息的过程[1]. 目前, 数据挖掘技术广泛应用于金融、医疗保健、市场、零售、制造、司法、工程和科学等领域. 将数据挖掘技术应用于气象预报中[2], 分析得出各种气象因子之间存在的关联模式, 从而降低分析气象数据的难度, 对建立更加精确的统计预测模型有了积极地帮助.

本文针对传统的天气预报统计方法存在的局限性, 提出了使用改进 FP-gowth 算法挖掘气象因子之间存在的关联. 以上海和北京的气象数据进行仿真实验, 实验结果表明, 改进的算法大大提高了天气预报的准确性.

2 改进的FP-growth算法介绍

2.1 Apriori 算法

关联规则的发现是数据挖掘的一个基本问题[3],

① 基金项目:国家自然科学基金(61502298)

收稿时间:2016-02-06;收到修改稿时间:2016-03-22 [doi:10.15888/j.cnki.csa.005407]

挖掘频繁模式集是关联规则的一个重要步骤. 基于频繁模式集发现的经典算法是 Apriori 算法[4]. 它是由 Ramakrishnan Srikant 和 Rakesh Agrawal 在 1994 年提出来的[5]. Apriori 算法的思想是: (1)一次扫描数据库, 对每个数据项进行计数, 得出 1 候选项集, 然后由定义的最小支持度筛选得出 1 频繁项集. (2)由 1 频繁项集进行连接得出 2 候选项集. (3)扫描数据库, 由 2 候选项集进行剪枝得出 2 频繁项集. (4)重复(2)、(3), 得出 k 频繁项集. (5)当 k+1 频繁项集为空时算法结束.

Apriori 算法的经典应用就是大家所熟知的“啤酒与尿布”的故事. 然而 Apriori 算法的缺陷也很明显: 算法扫描数据库的时间复杂度为 O(*k*), 其中 *k* 为产生的频繁项集. 当原始数据量太大时, 可能会产生大量的候选项集, 并且算法需要多次扫描数据库, 磁盘 I/O 次数太多, 效率比较低下.

2.2 FP-growth 算法

为了解决 Apriori 算法存在的缺陷, 韩家炜在 2000 年提出了基于挖掘频繁模式树的 FP-growth 算法[6]. FP-growth 算法采用分而治之的策略, 算法中采用了一种称为频繁模式树的数据结构. 它的基本思想是: 一次扫描数据库, 由长度为 1 的频繁模式开始, 构造它的频繁模式基, 然后由频繁模式基构造它的条件 FP 树, 最后递归地在该树上挖掘. 其中模式增长通过后缀模式与由条件 FP 树产生的频繁模式连接实现. FP 树的结构如图 1 所示. FP-growth 的优点是: 相对于 Apriori 算法, FP-growth 算法扫描数据库的时间复杂度为 O(1), 在实际应用中大大提高了挖掘的效率.

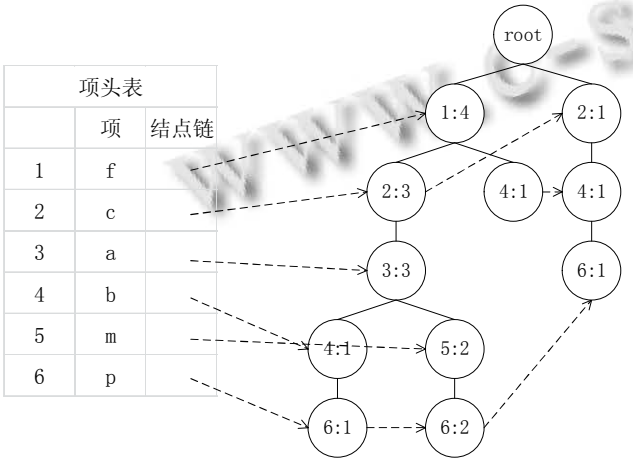

图 1　FP 树结构

FP-growth 算法可以广泛应用于市场、零售、医疗诊断、生物科学等各个领域. 通常, FP-growth 算法只适用于单维数据. 然而天气数据是多维数据, 如何将多维数据转化为单维数据, 本文提出了使用划分区间的方法, 可以有效地将多维天气数据转化为单维天气数据.

2.3 FP-growth 算法的不足

虽然相对于 Apriori 算法, FP-growth 算法在挖掘的效率上大大提高了. 但是在实际应用中, 依然存在着缺陷. 依据经验所得, 假定 min-support=0.5, min-confidence=0.6, 挖掘出的两条规则分别是(买咖啡, 买茶)support=0.56,confidence=0.7; (买咖啡, 不买茶)support=0.6,confidence=0.8; 可以清楚地发现, 两条规则都满足挖掘的条件, 可是实际情况中, 咖啡与茶是相互对立的两个事件, 买了茶的用户往往就会减少对咖啡的购买, 同样, 买了咖啡的用户就会减少对茶的购买量. 所以, 如何有效地排除这种有误导性的关联规则, 这是一件亟待解决的问题.

2.4 改进的 FP-growth 算法

针对挖掘出的关联规则有误导性的问题, 本文提出了通过关联度改进策略, 通过独立性检验可以较好地解决这个问题. 在 FP-growth 算法挖掘出强关联规则后, 使用独立性检验来进一步判断事务之间是否存在关联, 实验证明, 改进的算法提高了挖掘的准确性. 独立性检验的基本概念: 假设事件 A 与事件 B, 要求判断 A 与 B 之间是有关联还是相互独立.

(1) 首先建立列联表如表格 1 所示, 其中 A 与 B 的值域分别为{*A*1,*A*2}, {*B*1,*B*2}.

表 1　列联表

	B1	*B2*	总计
A1	*a*	*b*	*a+b*
A2	*c*	*d*	*c+d*
Sum	*a+c*	*b+d*	*a+b+c+d*

(2) 通过列联表计算公式其中 *a,b,c,d* 为样本, $n=a+b+c+d$ 为样本容量, 若 S^2 的值越大, 说明“事件 A 与事件 B 有关联”成立的可能性越大.

(3) 当 *a,b,c,d* 的值不小于 5 时, 查阅如下表格 2, 判断得出结论.

表 2　概率查询表格

$P(S^2 \geq S)$	0.50	0.25	0.1
S	99.33	109.14	118.50
$P(S^2 \geq S)$	0.05	0.025	0.01
S	124.34	129.56	135.81

例如: 当“A与B有关系”的S^2变量的值为125.635, 根据表格, $124.34 \le S^2 \le 129.56$, 所以“A 与 B 有关系”成立的概率为 1-0.05=0.95, 即“A 与 B 有关系”的概率为95%.

3 改进FP-growth算法在气象领域中的应用

3.1 数据的预处理

本文采用的气象数据是上海地区 2010-2014 近 5 年的气象数据, 首先进行了数据的清洗与处理. 获得的部分原数据如表 3 所示. 由于关联规则通常适用于单维的数据源, 原始的数据源显然无法适用, 本文利用了将多维的数据源转化为单维的思想[7], 基本思路是: 采用划分区间的思想, 将单个事务划分为多个事务. 每一条记录采用布尔型数据格式(1 代表是, 0 代表否), 在原始数据中: k 代表记录数, a 代表最高温度, b 代表最低温度, c 代表天气情况, d 代表风向, e 代表风力. 在数据处理过程中: 将温度、天气、风向、风力划分为多个区间, 其中在温度方面: 以十个单位为一个划分区间. 在天气部分: 分为晴天、阴天和雨天, 多云和阴天同属于阴天. 在风向方面: 以方位划分. 在风力方面: 以一个单位为划分区间. 在处理数据过程后: n 代表列数, k 代表记录数. 原始数据与处理后的数据的转换公式是:

$$\alpha=\begin{cases}1 & 10(n\text{-}1)\le\alpha\le 10n \quad n=1,2,3,4\\ 0 & \text{其它} \quad n=1,2,3,4\end{cases} \tag{1}$$

$$b=\begin{cases}1 & 10(n\text{-}6)\le b\le 10(n\text{-}5) \quad n=5,6,7,8,9\\ 0 & \text{其它} \quad n=5,6,7,8,9\end{cases} \tag{2}$$

$$c=\begin{cases}1 & c\in \text{晴}\vee\text{雨}\vee\text{阴(多云)} \quad n=10,11,12\\ 0 & \text{其它} \quad n=10,11,12\end{cases} \tag{3}$$

$$d=\begin{cases}1 & d\in \text{东}\vee\text{南}\vee\text{西}\vee\text{北}\vee\text{东北}\vee\text{东南}\vee\text{西北}\vee\text{西南}\\ 0 & \text{其它}\end{cases}$$
$$n=13,14,15,16,17,18,19,20$$
$$n=13,14,15,16,17,18,19,20 \tag{4}$$

$$e=\begin{cases}1 & n\text{-}17\le e\le n\text{-}18 \quad n=21,22,23,24\\ 0 & \text{其它} \quad n=21,22,23,24\end{cases} \tag{5}$$

处理后的部分结果如表格 4、5 所示.

表 3 处理前的部分数据

记录数 (k)	最高气温 (a)	最低气温 (b)	天气 (c)	风向 (d)	风力 (e)
1	11	3	多云	西风~西北风	3-4 级
2	12	8	小雨	南风~北风	3-4 级
3	16	4	晴	西风	4-5 级
4	28	15	晴转多云	西南	3-4 级
5	30	13	晴	南风	3-4 级
6	24	14	晴转多云	东南	小于 3 级
7	39	30	多云转雷阵雨	南风~西南风	小于 3 级, 3-4 级
8	5	-2	晴转多云	西北风	3-4 级
9	7	0	多云转晴	北风~东北风	3-4 级
10	5	-1	阴	北风	4-5 级

表 4 处理后部分数据

n(列数) / k (记录数)	1	2	3	4	5	6	7	8	9	10	11	12
1	0	1	0	0	0	1	0	0	0	0	1	0
2	0	1	0	0	0	1	0	0	0	0	0	1
3	0	1	0	0	0	1	0	0	0	1	0	0
4	0	0	1	0	0	0	1	0	0	1	0	0
5	0	0	1	0	0	0	1	0	0	1	0	0
6	0	0	1	0	0	0	1	0	0	1	0	0
7	0	0	0	1	0	0	0	1	0	0	1	0
8	1	0	0	0	1	0	0	0	0	1	0	0
9	1	0	0	0	0	1	0	0	0	1	0	1
10	1	0	0	0	1	0	0	0	0	0	1	0

表 5　处理后的部分数据

n(列数) / *k* (记录数)	13	14	15	16	17	18	19	20	21	22	23	24
1	0	0	1	0	0	0	0	0	0	1	0	0
2	0	1	0	0	0	0	0	0	0	1	0	0
3	0	0	1	0	0	0	0	0	0	0	1	0
4	0	0	0	0	0	0	1	0	0	1	0	0
5	0	1	0	0	0	0	0	0	0	1	0	0
6	0	0	0	0	1	0	0	0	1	0	0	0
7	0	1	0	0	0	0	0	0	1	0	0	0
8	0	0	0	0	0	0	0	1	0	1	0	0
9	0	0	0	1	0	0	0	0	0	1	0	0
10	0	0	0	1	0	0	0	0	0	0	1	0

3.2 实验环境

本文采用的实验设备是：PC 机一台(8G 内存，windows7 系统)，matlab 仿真软件．设置 min-support=0.5, min-confidence=0.6，进行 matlab 仿真测试．

3.3 结果与分析

算法对全年的数据和四季的数据分别进行了挖掘，由于本文主要研究温度、天气、风向、风力等几个因子之间的关联，所以选取了部分相关的挖掘结果．实验的部分结果如表 6、7 所示．

表 6　全年挖掘的部分结果

字段	频繁项集	频次
1	[1,6,11,23]	920
2	[4,8,10,21]	953
3	[4,8,10,24]	942
4	[2,6,11,22]	937
5	[3,7,10,22]	983
6	[3,7,11,22]	932

表 7　四季挖掘的部分结果

字段	频繁项集	频次
1	[3,6,22]	925
2	[4,8,12]	943
3	[3,7,10]	946
4	[1,5,11]	929
5	[1,6,17]	951
6	[1,6,22]	933

1) 规则 (1,6,11,23) 得到：上海全年阴雨天气时，最高气温和最低气温都偏低，风力也比较强劲．

2) 规则 (3,6,22) 可以得到：春季气温一般在 10℃~20℃，比较温和，风力 3-4 级，刮北风．春天虽然温度回升，但是还是会刮凛冽的北风．

3) 规则 (4,8,12) 可以得到：夏季雨水偏多，气温在 30 摄氏度以上，比较炎热．由于上海靠海，这个季节就是人们熟知的梅雨季节，人们要注意做好衣物的防霉措施．同时经常带着晴雨伞出门．这个季节也是台风常发生的季节，大家要做好防台风的措施．

4) 规则(3,7,10)可以得到：在秋季，温度一般在 10℃~20℃，比较凉爽．晴天居多．

5) 规则(1,5,11)可以得到：在冬季，气温一般在 0℃~10℃，比较寒冷，阴天比较多，所以这个时段天气会比较干燥．

6) 规则(4,8,10,21)得到：天气晴朗时，最高气温和最低气温都比较高，风力比较低．规则(4,8,10,24)得到：天气晴朗时，最高气温和最低气温都比较高，风力却比较高．这时两条规则就产生了矛盾．针对两条矛盾规则，使用独立性检验来解决这个问题．

首先建立列联表，如下表 8 所示．

表 8　天气列联表

	风力(3-4 级)	风力(4-5 级)	总计
晴	780	160	940
阴雨	420	480	900
总计	1200	640	1840

经计算所得：

$S^2=1840\times(780\times480-160\times420)^2/(940\times900\times1200\times640)\approx267.26$，参照表格 2，由($S^2\geq135.81)\leq0.01$，也就是说 S^2 的值大于 135.81 的概率非常小(只有 0.01),所以有 99%以上的把握认为晴天与风力小于 3-4 级有关．在实际情况中，上海天气晴朗时，风力一般都比较弱．所以，改

进的算法符合实际情况.

3.4 北京气象数据的挖掘

为了验证改进算法的准确性, 采用改进的 FP-growth 算法对 2001-2012 年北京天气数据进行数据挖掘. 挖掘的规则结果是:

1) 北京全年雨水天气时, 气温比较低, 风力比较强劲.

2) 北京春季温度比较温和(10℃~20℃), 风力比较大. 能见度比较好(10km 以上).

3) 北京夏季雨水天气时, 温度比较高(20℃~30℃), 常常会有暴雨天气, 能见度比较低(10km 以下).

4) 北京秋季温度相对于春季偏低, 风力一般, 能见度比较好.

5) 北京冬季气候温度偏低(-0℃~10℃), 晴天比较多, 能见度相对于夏季有所好转.

6) 北京冬季气候温度偏低(-0℃~10℃), 晴天比较多, 能见度比较差.

这里规则 5)与规则 6)也产生了矛盾, 采用改进的策略进行误导性规则的排除. 保留了规则 5), 由于北京市典型的大陆性气候, 能见度与湿度有着密切的关联, 水汽的扩散会造成能见度的降低. 所以规则 5)是符合实际的气候特点. 这也证明了改进的 FP-growth 算法具有普遍的适用性.

3.5 预测与分析

将关联规则算法应用于气象预报中, 对气象情况做短期预报, 这里使用 2013 年的上海气象数据源和 2011 年北京气象数据源进行短期气象的预测, 如表 9、10 所示.

表 9 上海气象数据源

日期	最高气温 (*a*)	最低气温 (*b*)	天气 (*c*)	风力 (*e*)
2013-07-06	31	26	大雨转小雨	4-5 级
2013-07-07	34	26	小雨	3-4 级
2013-07-08	27	21	大雨转暴雨	4-5 级
2013-07-09	33	27	多云转阵雨	3-4 级
2013-07-10	37	29	多云	3-4 级
2013-08-07	24	14	晴转多云	小于 3 级
2013-08-08	39	30	多云转雷阵雨	3-4 级
2013-08-09	36	29	晴转多云	3-4 级
2013-08-10	39	27	多云	小于 3 级
2013-08-11	34	28	阵雨	3-4 级
2013-08-12	35	27	多云	3-4 级

表 10 北京天气数据源

日期	最高气温(℃)	最低气温(℃)	天气(℃)	能见度(km)
2011-08-13	31	24	阴~雷阵雨	7.56
2011-08-14	30	21	中雨~大到暴雨	9.5
2011-08-15	27	22	阵雨~中雨	8.69
2011-08-16	33	27	多云转阵雨	6.92
2011-08-17	32	23	晴	8.69
2011-12-10	-5	5	晴	11.11
2011-12-11	-4	8	晴	10.46
2011-12-12	-3	7	晴转多云	10.78
2011-12-13	-4	4	多云转晴	11.91
2011-12-14	-6	1	晴	11.75

从表 9 中的数据可以看出: 2013-07-06~2013-07-10 和 2013-08-07~2013-08-12 的天气阴雨天气比较多, 阴雨天气风力比较强劲, 晴天时风力比较低. 实际天气情况和改进的算法得出的预测结果比较符合.

从表 10 中的数据可以看出: 2011-08-13~2011-08-17 北京温度比较高, 雨水天气比较多, 能见度比

较低. 2011-12-10~2011-12-14 北京的温度比较低, 晴天比较多, 能见度相对于夏季有所好转. 实际天气情况和改进后的算法得出的预测结果比较符合.

3.6 算法比较

本文将改进前的算法和改进后的算法分别进行挖掘, 挖掘的规则数如下图 2, 3, 4, 5 所示, 结果表明, 使用改进后的挖掘算法, 产生的规则数明显减少. 经计算所得, 采用上海和北京的数据源, 改进后的算法相对于改进前的算法规则数分别减少了 20%左右和 30%左右. 改进后的算法准确率有了明显的提高.

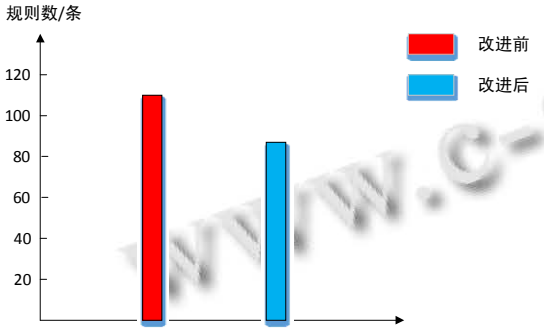

图 2 全年挖掘数据(上海)

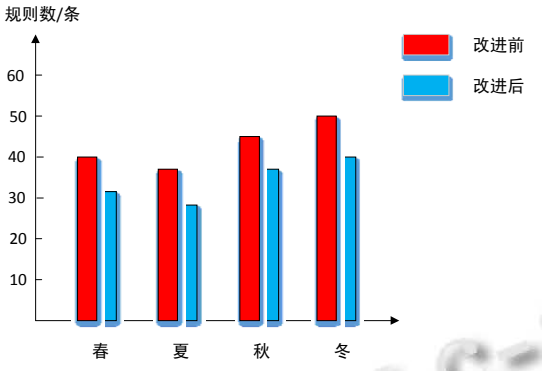

图 3 四季挖掘数据(上海)

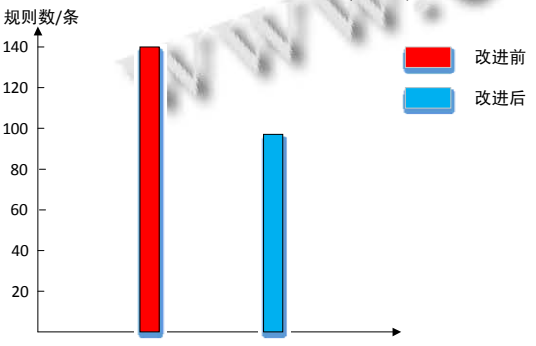

图 4 全年挖掘数据(北京)

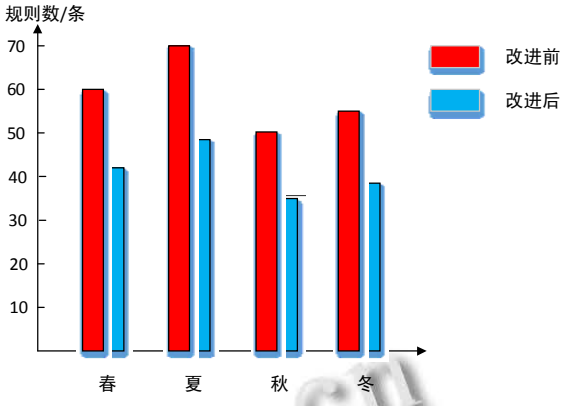

图 5 四季挖数据(北京)

4 结语

本文提出了改进的 FP-growth 算法进行天气数据的挖掘, 针对挖掘结果可能产生的误导性的问题, 提出了独立性检验的改进策略. 以两种典型的气候类型城市上海和北京的气象数据进行了仿真实验, 仿真结果表明, 改进策略能够明显地排除有误导性的规则, 降低了分析气象数据的难度, 对天气预报的准确度有了显著的提高.

参考文献

1 Han JW, Kamber M. 范明,孟小峰,译.数据挖掘概念与技术. 第 3 版.北京:机械工业出版社,2012.

2 赵鹏倪,志伟,贾兆红.利用数据挖掘技术从气象数库中建立范例库.微机发展,2002,12(3):67–70.

3 谢娜,戚晓明,朱洪浩,郭有强.半结构化多 Web 文本数据挖掘的研究.齐齐哈尔大学学报(自然科学版),2015,(2):75–78.

4 Liu HT, Guo RX, Jiang H. Research and improvement of Apriori algorithm for mining association rules. Computer Application and Software, 2009, 26(1): 146–149.

5 Agrawal R, Srikant R. Fast algorithms for mining association rules in large database. Proc. of the 20th International Conference on Very Large Databases, 1994, 23(3): 487–499.

6 Grahne G, Zhu J. Fast algoritms for frequent itemset mining using FP-trees. IEEE Trans. on Knowledge and Data Engineering, 2005, 17(10): 1347–1362.

7 王德兴,胡学钢,刘晓平,王浩.改进购物篮分析的关联规则挖掘算法.重庆大学学报(自然科学版),2006,29(4):105–107.