实验 5 频繁项集挖掘

更新日期：2018-11-15

1. 实验要求

实验背景

频繁项挖掘或关联规则挖掘问题是数据挖掘中的基本问题，一个经典实例就是购物篮问题。超市或者网店可以根据顾客的购买记录进行频繁项或关联规则挖掘，从而发现客户的购买习惯。比如说购买产品 X 的同时购买 Y，于是可以根据这种购物习惯进行货架调整，将关联度高的商品放在一起以提高销售量。频繁项挖掘的基本概念请参考 wikipedia 关联规则学习。

实验任务

Apriori 算法是频繁项集挖掘中的经典算法。Apriori 算法通过多轮迭代的方法来逐步挖掘频繁项集。在第一轮迭代中，计算事务数据库中每一项的支持度并找出所有频繁项。在之后的每轮迭代中，将前一轮生成的频繁 k-项集作为本轮迭代的种子项集，以此来生成候选(k+1)-项集。这些候选集在整个事务数据库中可能是频繁的，也可能是非频繁的，在本轮迭代中，需要计算每个候选人(k+1)-项集在事务数据库中的实际支持度，以找出全部(k+1)-频繁项集并将其作为下一轮的种子项集。这样的迭代过程将一直进行下去，直到不能产生新的频繁项集为止。

请在 Spark2.3 平台上实现 Apriori 频繁项集挖掘的并行化算法。要求程序利用 Spark 进行并行计算。本次实验要求在经典数据集上计算出极大频繁项集。

2. 输入输出

输入要求

输入数据为一个文件和最小支持度。该文件由若干行组成，每一行代表一个事物，由以空格分隔的整数组成：

A B C D ….

下面的框中是文件部分内容的示例：

```
33 44
21 22 45 67
21 22 78 89 90 91
24 22 78 89 99 91 97 13
```

最小支持度为 0-1 以内的小数。

输出要求

请输出在最小支持度 min_supp 下的所有极大频繁项集（包括给出极大频繁项集的项数及其支持度）。请在实验结果压缩包中包含所有结果。

其他要求

本次实验要求自己动手搭建 Spark 环境（可采用单机伪分布式模式）。语言可使用 python，scala，java 等 Spark 支持的语言。

3. 实验报告要求

请在报告中报告如下内容：

实验设计说明，包括主要设计思路、算法设计、程序和各个类的设计说明：

- 本地 Spark 环境的搭建说明及其截图。
- 基于 Spark 的 Apriori 并行算法设计思路。
- 算法的伪代码（或者带注释的实际代码，如果使用实际代码，请做好排版）。
- 最终统计出的极大频繁项集及其支持度，输出结果文件的截图 。
- 程序运行性能分析。
- 性能、扩展性等方面存在的不足和可能的改进之处。
- 源代码、可执行程序 JAR 包（.py 文件）、JAR 包(.py 文件) 运行方式说明（助教可能会在集群上重新执行 JAR 包）。
- 请在报告中包含在集群上执行作业后，WebUI 执行报告的内容。请完整包括执行报告内容，否则影响分数。整个程序运行中的 Spark Job，需要附上一个相应的 WebUI（默认为 localhost:4040 端口）执行报告。

4. 选做内容

该部分内容不做要求，供学有余力的同学尝试练习。

选做 1: 本次实验中，我们求取了极大频繁项集，请参考最小置信度的概念，根据结果设置合适的最小置信度生成关联规则，最小置信度为输入值。

选做 2: Apriori 算法的缺点：(1) 由频繁 k-1 项集进行自连接生成的候选频繁 k 项集数量巨大。(2) 在验证候选频繁 k 项集的时候需要对整个数据库进行扫描，非常耗时，请基于以上缺点进行改进优化。（如果实验结果性能提高较大，有良好的性能分析，会进行补偿性加分，但总分不会超过本次实验的满分）。

5. 实验数据

请分别计算以下数据集的所有极大频繁项集（包括给出极大频繁项集的项数及其支持度）。

1. http://fimi.ua.ac.be/data/chess.dat **最小支持度 min_supp=0.8**
2. http://fimi.ua.ac.be/data/mushroom.dat **最小支持度 min_supp=0.8**

3. http://fimi.ua.ac.be/data/connect.dat 最小支持度 min_supp=0.9

友情提示

本次实验可能会比较耗时，请尽早开始实验，以免最后非常仓促。