实验三

3.1 实验要求

数据集：某中文 IT 技术社区用户行为日志数据

本次实验所使用的数据集由某中文 IT 技术社区提供，共包含 157,427 位用户 在 2015 年期间产生的多种类型的行为数据。数据集基本统计信息如下：

数据类别	数据内容（行为类型）	数据量
用户行为数据	用户发表博客记录	1,000,000 条
	用户浏览博客记录	3,746,161 条
	用户评论博客记录	182,273 条

数据集中用户编号为 U0000001 至 U0157247，文档编号为 D0000001 至 D1000000。实验数据由六张表构成，分别是“用户基本信息表”、“用户教育情况表”、“用户技能表”、“用户兴趣表”、“用户行为表”和“文章统计信息表”。数据表以文本文件的形式存储，各字段之间以“\001”分割。各数据表的说明详见“数据仓库课程实验三数据说明”。

要求介绍：

1) 统计并展示一段时间内的用户行为数据。查询维度包括：时间跨度（天级别）、用户等级、用户学历、行为类型等。

2) 统计并展示一段时间内的文章活跃数据。查询维度包含：时间跨度（天级别）、博文板块、操作类型等。

3) 统计并图形化展示某天领域活跃程度。展示指标包含：各领域用户分布、各领域文章活跃度（浏览量）、各教育等级所在领域的占比等。

4) 周报表生成包括：统计一周内各小时用户活跃（行为）的期望和标准差、统计每天用户活跃最热和最冷的小时以及活跃程度、统计最活跃的前 10 名用户、统计最活跃的前 10 领域。（存储至 HDFS）

5) 选定或者设计某一个统计需求（要求统计粒度至少为天级别），分别针对 hive 中的小时粒度和 MySQL 中的天粒度两个数据集进行统计。进行时间消耗、资源消耗的对比。

备注：本实验总共至少需要完成三个实验点。第一个实验点可以从前三个要求中至少选择一个；第二个实验点是要求 4）；第三个实验点是要求 5）。其中，以上要求中需要统计的维度如果出现数据空缺，一律按照“无”这个维度统计，比如用户学历。

3.2 实验目的

实验二主要利用了单机程序进行数据的实时解析与存储，出现了实时解析处理和存储等效率瓶颈问题。本实验将使用分布式实时处理与分布式存储技术解决这些问题，同时也将尝试利用分布式处理技术对结构化的用户行为日志数据进行一些常用的统计分析处理。主要实验目标有：

1) 利用分布式实时处理框架实现数据实时装载；
2) 按照不同时间粒度，利用分布式处理技术对实时解析出的结构化用户行为日志数据进行聚合处理，并存储到 HDFS、HBase、Hive 或者 MySQL 中以便进一步分析处理；
3) 尝试将较粗时间粒度的用户行为日志数据聚合结果，利用基于 Web 的方式进行可视化展示与查询等；
4) 对同一需求分别实现从不同粒度数据集进行统计计算，体会数据仓库中多层数据模型设计的必要性；
5) 尝试设计一套带有反馈机制的闭环业务系统（可选加分项），如推荐系统等。

3.3 实验环境

1) MySQL；
2) Kafka 集群；
3) 分布式文件系统：HDFS ；
4) 数据仓库：Hive；

5) 分布式数据库：HBase；

6) 分布式近实时处理框架：Spark Streaming；或分布式流式处理框架：Storm；

7) 分布式计算框架：Hadoop MapReduce 或 Spark 等；

8) 编程语言：Java（推荐使用）或 Scala 或 C++等；

3.4 实验内容

- 分析用户行为日志数据，设计用户行为日志数据的细节存储模型（HBase）。根据需求设计多粒度、多维度的数据仓库。要求 Hive 包含小时、天粒度数据。设计 MySQL ER 模型，存储天粒度数据和用户信息、博客信息等静态数据。提示：小时聚合、天聚合的时候会用到静态数据，所以先设计存储 MySQL 中的静态数据。
- 设计 MySQL ER 模型，存储用户信息、博客信息等静态数据。
- 编写生产者程序，将用户行为日志数据按照时间顺序发送至 Kafka，并通过实时计算框架 Storm 或近实时计算框架 Spark Streaming 解析原始日志数据并存储至 HBase。
- 利用分布式计算框架 Spark 或者 MapReduce 从 HBase 读取细节数据，并聚合成小时粒度数据存储至 Hive。同样利用 Spark 或者 MapReduce 从小时粒度聚合天粒度并存储 MySQL。
- 根据需求设计前台网页交互，推荐 Java Web，按照需求指定的查询方式，从 MySQL 中查找数据并展示。
- 根据周报表需求，编写程序从 Hive 小时粒度数据中统计形成报表文件，存储在 HDFS 上。

3.5 实验步骤

1) 分析数据和需求，设计存储模型。包括，HBase 细节数据存储模型，Hive 小时、天粒度存储模型，MySQL 存储模型（包括天粒度数据存储模型和静态数据存储模型）；

2) 编写 Kafka 生产者，发送用户行为日志数据；利用 Spark Streaming 或者 Storm 编写 Kafka 消费者，并解析数据存储至 HBase；

3) 编写小时聚合程序，从 HBase 读取细节数据并统计成小时粒度数据存储至 Hive。编写天聚合程序，读取 Hive 的小时粒度数据，并存储至 MySQL；

4) 编写 web 前端程序，满足查询需求；

5) 编写周报表生成程序，从 Hive 小时粒度读取数据并统计，最后存储至 HDFS；

6) 测试体会不同粒度数据集统计指标的差异性。（详见要求 5））

3.6 加分项（选做）

1) 根据应用背景，设计一个基于实时用户行为日志数据的实时推荐系统；

2) 设计并创建推荐系统各个功能模块所需要的概念模型、逻辑模型；

3) 可以参考设计一个以“周”为长度的滑动窗口进行分析，每次分析结果用于当下一小时内的文章推荐，即窗口的滑动步长为一小时；也可以自己寻找设计推荐算法实现一个实时推荐模块，为每个用户推荐感兴趣的相关文章；计算的指标建议存储在 Hive 中。

4) 对推荐结果进行基于 Web 的简单展示。

备注：主要目标是实现一个具有反馈机制的架构，实现数据流闭环，要求具有一定的结果展示。算法在本环节不是重点。