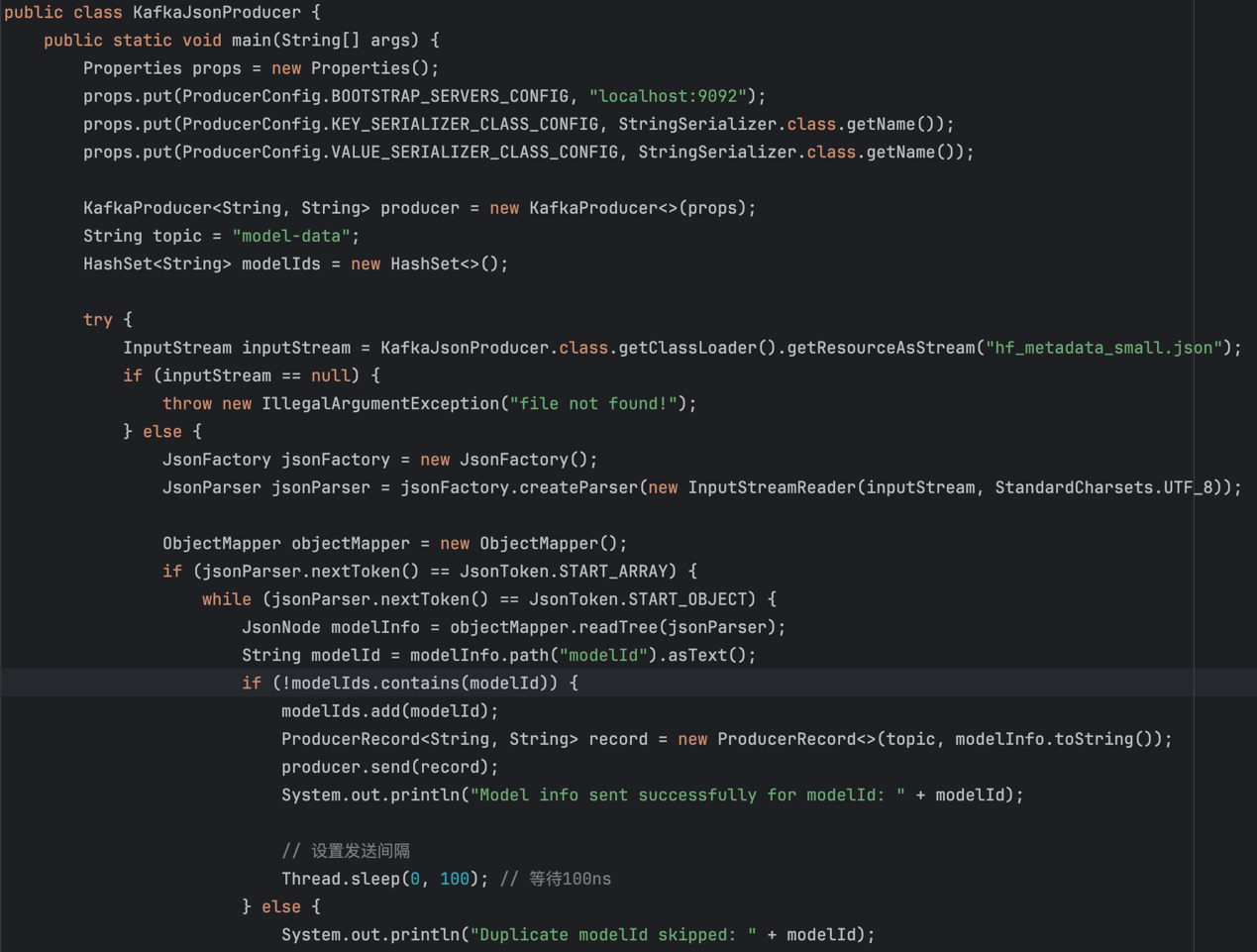
作业2-基于分布式大数据平台的数据分析服务开发

1. **数据处理方法**

本项目的核心是处理和分析机器学习模型的元数据。处理流程包括三个主要环节：数据采集、数据清洗与转换、以及数据存储。

1. 数据采集：

* 使用 KafkaJsonProducer 类，该类从 hf\_metadata.json 文件中读取JSON格式的数据，并将其作为消息发送到Kafka主题 model-data 中。
* 数据以流的形式持续推送到Kafka，确保实时性和连续性。



1. 数据流处理：

* 数据清洗：通过 DataCleansingBolt 对接收的数据进行预处理。

去重：确保不处理重复数据。

去除不合法的输入：检查和过滤掉不符合预期格式或内容的数据。

处理残缺数据：补全缺失的字段或删除残缺不全的记录。

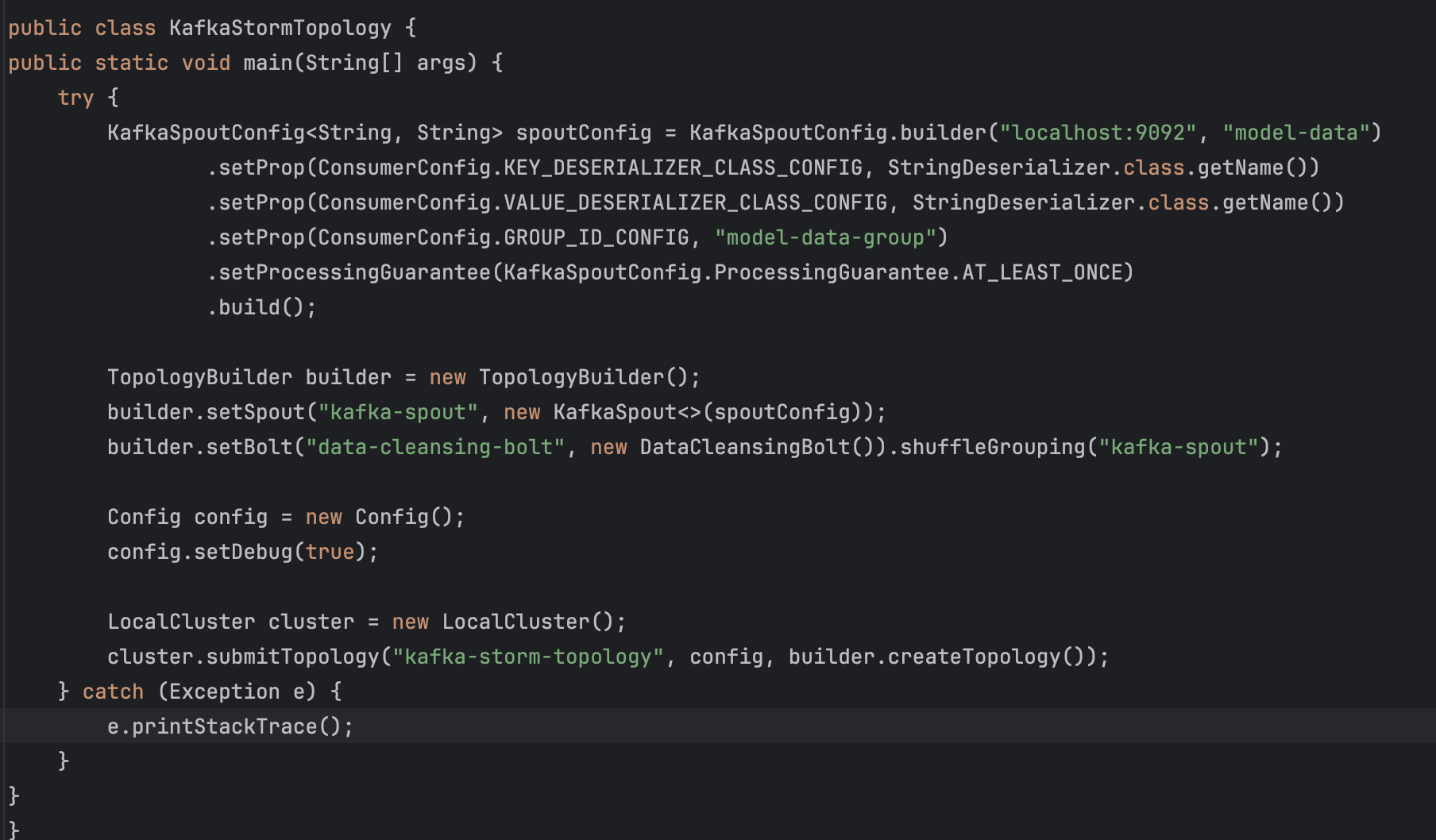
过滤掉点赞和下载次数过低的数据：基于 likes 和 downloads 过滤数据。

* 数据转换：在清洗过程中，对数据字段进行标准化和格式化，确保数据质量和一致性。



1. 数据存储：

* 创建一个拓扑，其中包含一个从Kafka主题获取数据的Spout和一个用于数据清洗的Bolt。通过将Spout和Bolt连接在一起，实现了从Kafka获取数据并进行清洗的功能
* 使用 HiveBolt 将清洗和转换后的数据存入Hive数据库，使用预定义的表结构和数据模式。

对storm通过流式数据处理得到的json文件进行处理，提取出格式化所需的维度，上传到hdfs中。提取的维度包括了：模型名称（modelId），模型最后修改时间，pipeline\_tag，模型作者，模型点赞数量，模型下载数量，library name，datasets，家族（architectures）和类型(model\_type)。

文本

描述已自动生成

先将提取出的数据上传到hdfs中

bin/hdfs dfs -put /cleaned\_data /user/hadoop/

再从hdfs中，通过load的方式将文件上传到hive数据库中进行存储

文本

描述已自动生成

1. **维度设计**

在Hive中，数据以以下维度存储，便于进行复杂的查询和深入的数据分析：



* **modelid**（模型名称）：模型的唯一识别名。
* **pipeline\_tag**（数据处理管道）：机器学习中使用的一种标记，便于处理不同流程
* **lastmodified**（最后修改时间）：模型最后修改时间
* **author**（发布作者）：模型的创建者或发布者。
* **likes**（喜爱数）：模型被喜爱点赞的次数。
* **downloads**（下载量）：模型被下载的次数，反映模型的流行度。
* **model\_type**（学习模型类别）：模型使用的机器学习或深度学习模型类别
* **datasets**（数据集）：模型训练使用的主要数据集。
* **library\_name**（模型库）：模型使用的库和框架。
* **architectures**（模型架构）：模型的整体结构或架构设计。

1. **数据处理**

使用pyspark和fineBI两种方式结合处理hive中的数据。

（1）Pyspark：创建sparkSession，读取hive中导出的csv的数据，对数据进行排序，导出前20条结果。

|  |  |
| --- | --- |
| 按模型家族排序： | 按数据集排序： |
|  |  |

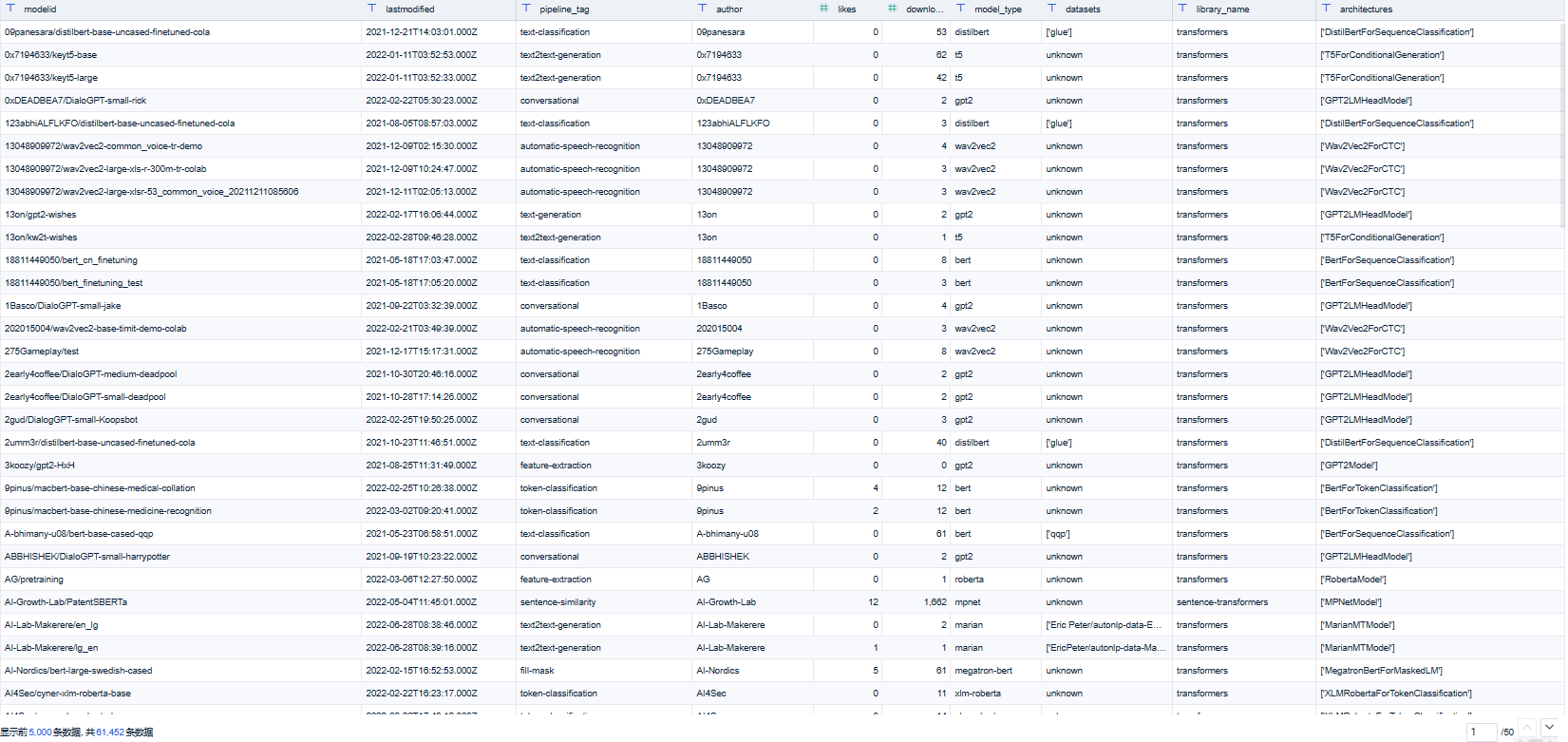
（2）fineBI：使用fineBI连接hive数据库，处理hive中的数据。

连接设置：



过程中需要云服务器对主机的浮动ip启用10000端口的tcp安全组。

查看数据：



对数据进行处理，包括：模型下载量排行，作者下载量排行，库被使用的数量。

1. **看板**

使用FineBI仪表盘制作看板。将处理后的数据选用合适的图表放置在仪表盘上。

