项目说明文档

三种数据模型比较及性能分析

——数据仓库项目

作 者 姓 名： 1754119 陈玉洁

1754197 刘岚心

1551531 吴国栋

指 导 教 师： 朱宏明

学院、 专业： 软件学院 软件工程

同济大学

Tongji University

目 录

[1 项目简介 1](#_Toc27904157)

[1.1 背景分析 1](#_Toc27904158)

[1.2 功能分析 1](#_Toc27904159)

[2 模型选择 2](#_Toc27904160)

[2.1 关系型数据仓库存储模型MySQL 2](#_Toc27904161)

[2.2 图数据库存储模型Neo4j 3](#_Toc27904162)

[2.3 分布式文件系统存储模型Hive 4](#_Toc27904163)

[3 性能分析 5](#_Toc27904164)

[3.1 关系型数据仓库存储模型MySQL 5](#_Toc27904165)

[3.2 图数据库存储模型Neo4j 5](#_Toc27904166)

[3.3 分布式文件系统存储模型Hive 5](#_Toc27904167)

[3.4 性能对比分析 5](#_Toc27904168)

[4 问题解决 5](#_Toc27904169)

[4.1 问题一： 5](#_Toc27904170)

[4.2 问题二： 5](#_Toc27904171)

[4.3 问题三： 5](#_Toc27904172)

[5 项目总结 5](#_Toc27904173)

# 1 项目简介

## 1.1 背景分析

本项目为2019年秋季数据仓库项目，主要考察大家对数据仓库的理解、数据爬取及清洗的能力、数据导入数据库的能力以及三种不同数据模型之间的对比，包括不同数据模型的优劣势，同一模型下不同数据量的查询性能比较。该项目的数据来源为亚马逊网站的电影信息及其用户评论文件。需要我们通过**数据爬取、数据清洗、数据导入、模型分析**几个阶段来提升对本项目的理解和实操能力。

亚马逊网站电影首页中共有超过50000条电影的信息及大小接近9G的用户评论信息，需要我们对这些数据进行处理，并导入三种不同的数据模型中，通过对常用语句的查询，横向纵向比较不同数据模型的优劣势以及在相同模型下，不同数据量的查询性能。

## 1.2 功能分析

本项目重点考察两个方面内容：**一是对数据的处理能力、二是数据模型的比较及调优。**通过制作简易的前端界面，项目需要达到以下功能要求：

（一）针对电影及其周边信息，建立基于关系型数据仓库、分布式文件型数据仓库系统和图数据库的数据仓库系统，并进行系统性能比对

1 、能够从数据来源处获取数据，进行数据预处理

2 、建立关系型数据仓库存储模型，存储数据

3 、建立分布式文件系统存储模型，存储数据

4 、建立图数据库存储模型，存储数据

5 、在数据展现的界面上能够执行数据应用中的查询，并将在三种不同存储模型上的执行时间以数值的方式和图表的方式显示在界面上

6 、对每种存储方式结合本项目说明各自适用于处理什么查询，针对本项目在存储优化中做了什么工作，优化前后的比较结果是怎样的，以文档的方式提交

（二）数据来源：数据来源自Snap的文本文件和Amazon网站，数据包括但不限于以下信息： 电影ID，评论用户ID，评论用户ProfileName，评论用户评价Helpfulness，评论用户Score，评论时间Time，评论结论Summary，评论结论Text，电影演员，电影上映时间，电影风格，电影导演，电影主演，电影演员，电影版本等信息。

（三）数据应用：常见查询及统计（占总查询数目）=80%）：

1 、按照时间进行查询及统计（例如XX年有多少电影，XX年XX月有多少电影，XX年XX季度有多少电影，周二新增多少电影等）

2 、按照电影名称进行查询及统计（例如 XX电影共有多少版本等）

3 、按照导演进行查询及统计（例如 XX导演共有多少电影等）

4 、按照演员进行查询及统计（例如 XX演员主演多少电影，XX演员参演多少电影等）

5 、按照演员和导演的关系进行查询及统计（例如经常合作的演员有哪些，经常合作的导演和演员有哪些）

6 、按照电影类别进行查询及统计（例如 Action电影共有多少，Adventure电影共有多少等）

7 、按照用户评价进行查询及统计（例如用户评分3分以上的电影有哪些，用户评价中有正面评价的电影有哪些等）

8 、按照上述条件的组合查询和统计

# 2 模型选择

## 2.1 关系型数据仓库存储模型MySQL

1、环境配置：

（操作系统、软件版本等）

2、选择理由：

MySQL是时下使⽤率最⾼的⼏款关系型数据库之⼀，且其体积相较其他关系型数据库更⼩巧且性能不输⼤型关系型数据库。关系型数据库是我们最常接触也是在对数据进⾏存储时会最先想到的数据库类型。我们想要借助关系型数据库以及⾏式存储对我们的数据进⾏存储，并通过对应的数据库操作对存储对数据进⾏分析/查询，实现我们对应的⽬的。

3、存储简介：

在雪花还是星型存储的问题上，小组内部进行了讨论，最终选择了雪花模型。原因如下：①雪花模型使用规范的数据、数据冗余较少。②虽然规范化存储会带来时间上消耗，但因为数据量不是特别大的缘故，两者查询速没有明显的差距。

在选择雪花模型的基础之上，我们对数据库设计了实体表和联系表，各个实体表有自己唯一的主码，实体表之间使用联系表进行关联，减少了很多实体数据的存储，符合第三范式。

## 2.2 图数据库存储模型Neo4j

1、环境配置：

macOS Mojave10.14.5

Neo4j Desktop1.2.3

2、选择理由:

①高性能:：Neo4j以图的遍历算法来帮助查询数据，查询时从⼀个节点开始，根据其连接的关系，快速和⽅便地找出它的邻近节点。这种查找数据的⽅法并不受数据量的⼤⼩所影响，因为邻近查询始终查找的是有限的局部数据，不会对整个数据库进⾏搜索。所以，Neo4j具有⾮常⾼效的查询性能，相⽐于RDBMS可以提⾼数倍乃⾄数⼗倍的查询速度。⽽且查询速度不会因数据量的增长⽽下降。

②灵活性：图数据结构的⾃然伸展特性及其⾮结构化的数据格式让Neo4j 的数据库设计可以具有很⼤的伸缩性和灵活性，使其可以随着需求的变化⽽增加的节点、关系及其属性并不会影响到原来数据的正常使⽤，因此在项⽬后期的推进中，我们也可以不断的快速修改neo4j数据库中的内容来满足我们的查询需求。

③直观性：图数据库使⽤图的形式作为数据库最主要的展现形式，可以更清楚的帮助我们理解整个数据库中数据之间的联系，Cypher语⾔的灵活性也帮助我们更轻松的操控数据库。

3、存储简介：

本项⽬中主要建⽴了Neo4j的两个不同的库，⼀个库是围绕电影的相关信息，我们在其中存储了和电影有关的所有信息，包括导演，演员，类别，语⾔，工作室等等，节点与节点之间通过不同的关系相连接。第⼆个库针对合作关系，存储了导演，演员，以及类别之间的联系， 通过节点与节点之间的关系，记录他们彼此的合作次数，类别的引⼊也帮助我们分析导演的执导风格。

## 2.3 分布式文件系统存储模型Hive

1、环境配置：

（）

2、选择理由：

Hive⾸先有很多以上数据库所不具有的优点，如扩展性和容错性，本项⽬我们选择hive来处理⼀部分数据主要是作为MySQL数据库的对照。针对我们项⽬ 的百万级的数据量来对⽐分析关系型数据库和分布式数据库在数据量较⼤时的性能优劣性，以此窥见数据仓库对⽐于数据库的所展现出来的优点。同时对于项⽬中的部分功能需求组合采⽤hive与其他数据库分治的⽅式，来实现复杂的功能需求，以此来学习⼯程中数据仓库与普通数据库结合的实现⽅法。⽽由于数据量及需求的限制，我们只可窥见数据仓库其作⽤的冰⼭⼀⾓，希望藉此加深我们对数据仓库的理解。

3、存储简介：

在hive中我们存储的数据与MySQL中⼀样。因此建⽴了与MySQL完全相同的存储结。

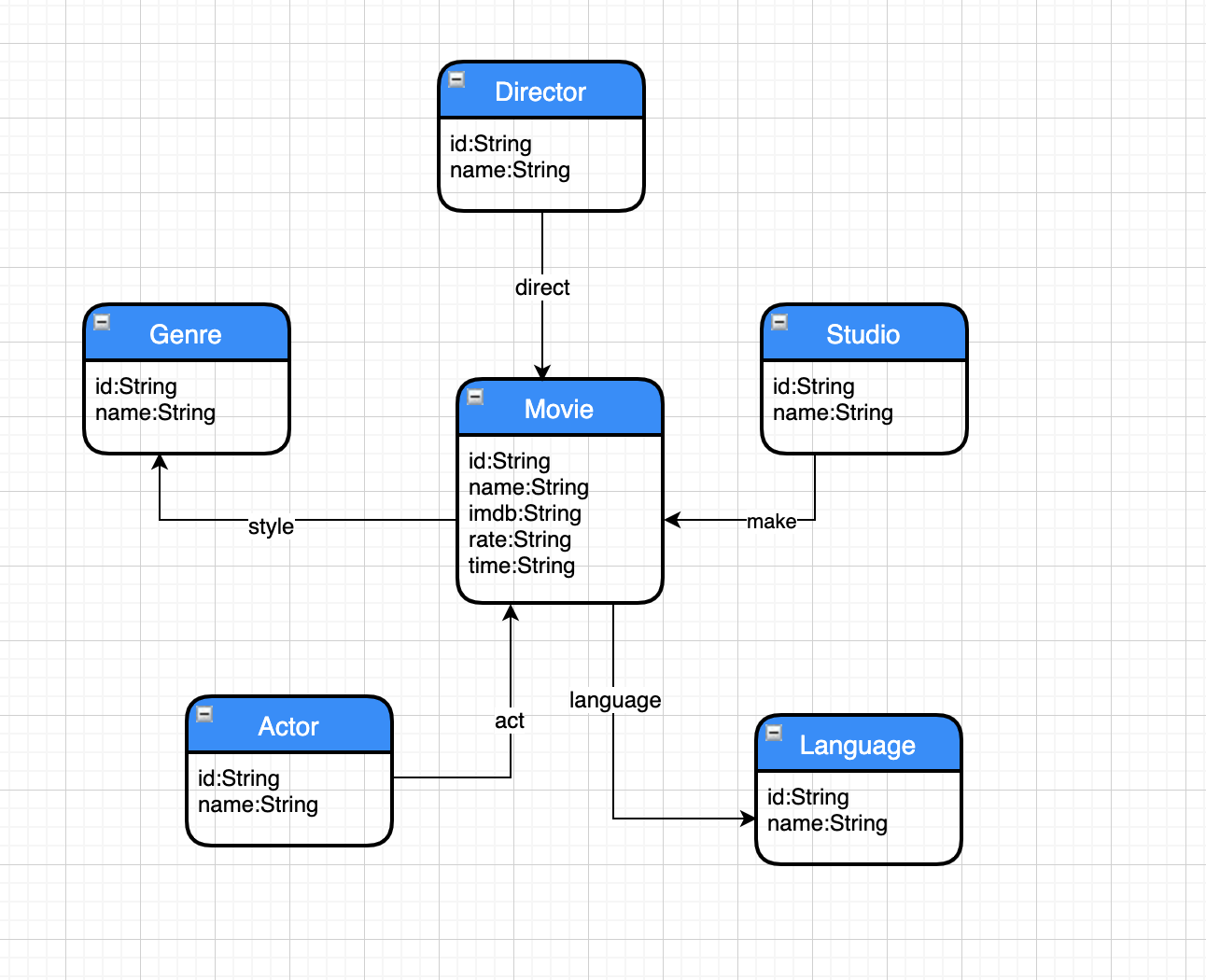
# 3 性能分析

## 3.1 关系型数据仓库存储模型MySQL

## 3.2 图数据库存储模型Neo4j

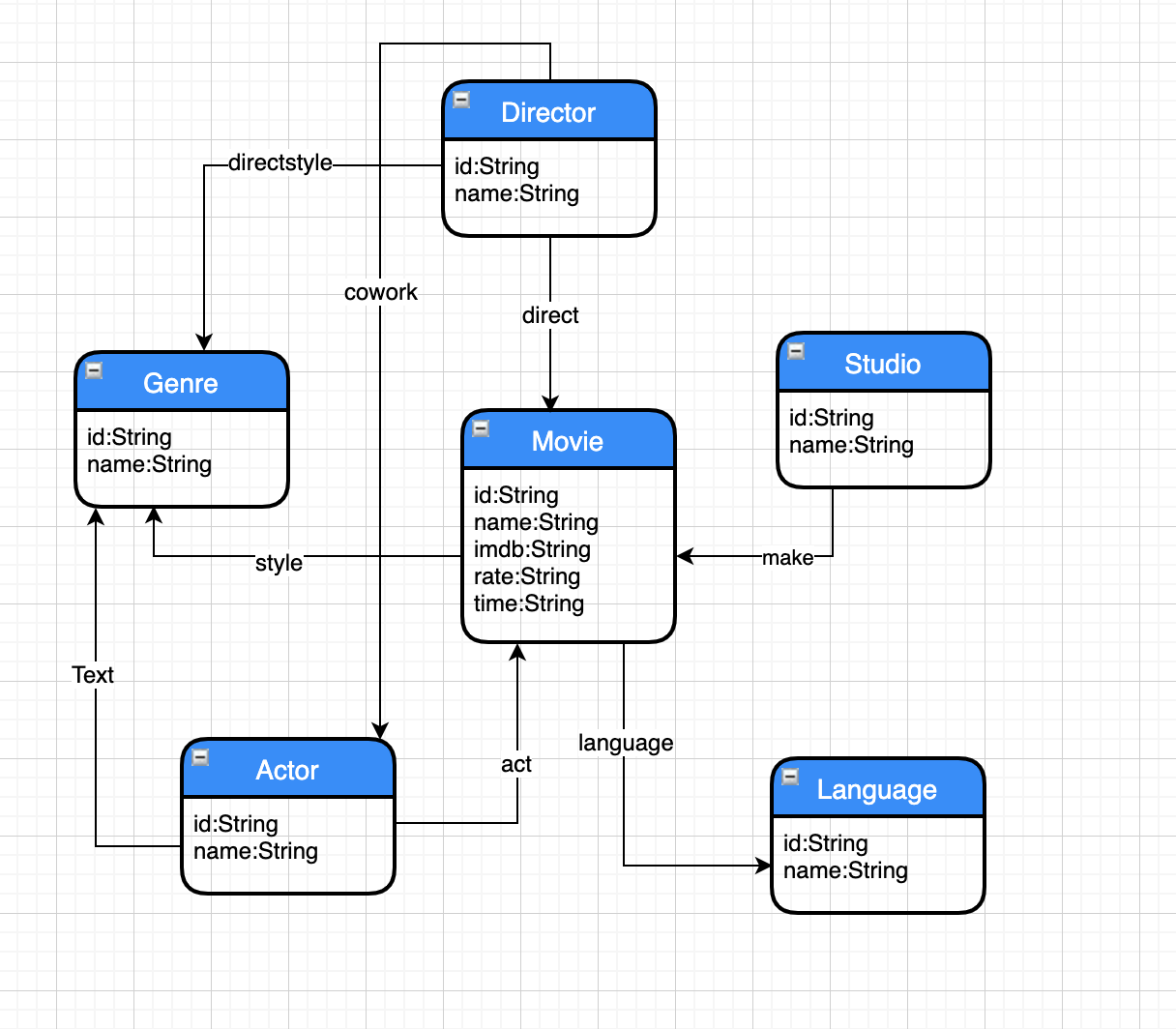
在Neo4j中，存储节点时，每个节点都有指向其邻居节点的指针，可以让我们在O(1)的时间内找到邻居节点。另外，按照官方的说法，在Neo4j中边是最重要的,是”first-class entities”，所以单独存储，这有利于在图遍历的时候提高速度，也可以很方便地以任何方向进行遍历。

本项⽬中为了对比性能，主要建⽴了两个Neo4j的库，⼀个库是围绕电影的相关信息，我们在其中存储了和电影有关的所有信息，包括导演，演员，类别，语⾔，工作室等等，节点与节点之间通过不同的关系相连接。第⼆个库在第一个库的基础上，针对合作关系，存储了导演，演员，以及类别之间的联系， 通过节点与节点之间的关系，记录他们彼此的合作次数，类别的引⼊也帮助我们分析导演的执导风格。



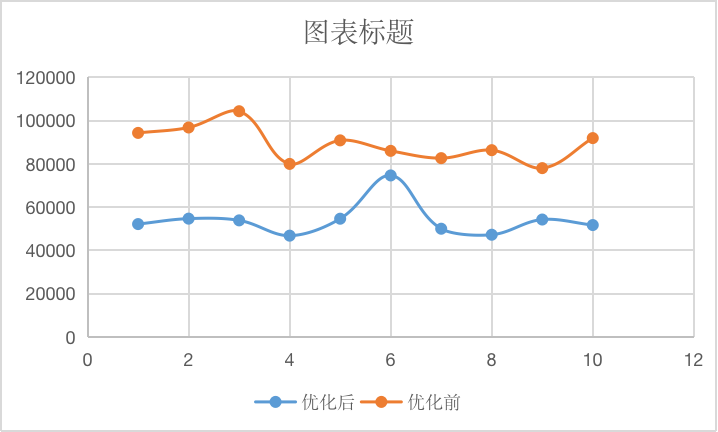
优化前

为了提高速度，采取了冗余存储。多加了一些边，速度有明显的提升。

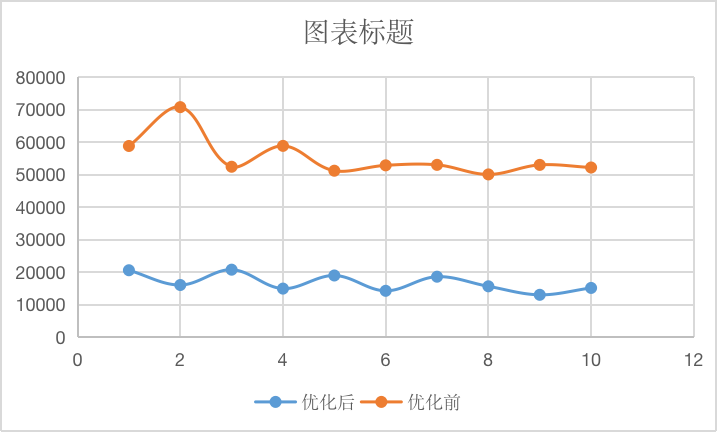


优化后

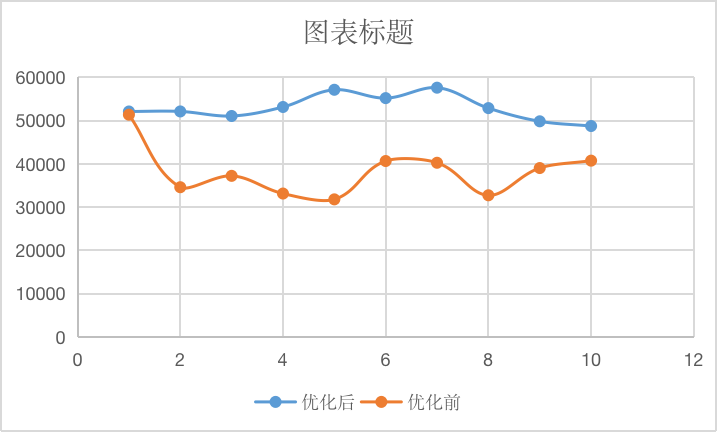
下图是某三条语句连续执行10次时，优化前和优化后耗时的比较。



1. 语句



1. 语句



1. 语句

通过图表可以明显看出优化前后的性能对比，也进一步证明了“在Neo4j中边是最重要的,是”first-class entities”，所以单独存储有利于在图遍历的时候提高速度”。

## 3.3 分布式文件系统存储模型Hive

## 3.4 性能对比分析

# 4 问题解决

## 4.1 问题一：

## 4.2 问题二：

## 4.3 问题三：

# 5 项目总结

* 本项⽬使⽤了JavaWeb框架，并基于sementicUI进⾏前端开发。
* 在数据库选择上，我们使⽤了Mysql，hive，Neo4j以及influxDB四个不同的 数据库进⾏横向纵向⽐对，通过实现⼀定的基本功能搜索以及多表联查，统计他们的性能，查询时间等数据并进⾏相应的分析，对于不同数据库的优劣势有了更为清晰的了解。
* 在项⽬过程中，我们将上课学到的知识应⽤到实践中，尝试了雪花，星型等不同的存储结构，并根据⾃⼰的项⽬实情选择了最适合我们的项⽬存储结构。针对不同的实验现象，我们也通过⽹络等资源来进⾏辅助学习，帮助我们更好的了解不同数据库以及其不同的存储，读取等⽅式。

项⽬过程中，特别感谢⽼师和学长们的帮助，让我们更为深⼊了解了数据仓库技术，为我们今后的项⽬实践打下了扎实的基础