



北京交通大学
Beijing Jiaotong University

软件构思综合训练

影视舆情分析系统 概要设计

组 长：汤新宇 17301137

小组成员：王心蕊 17301048

陈嘉琪 17301060

唐 麒 17301138

张钰铎 17301145

贡献分配：同 等 贡 献

指导老师：李 宇

北京交通大学软件学院

2020 年 6 月 15 日

目录

1	引言	1
1.1	编写目的	1
1.2	背景	1
2	总体设计	1
2.1	概述	1
2.2	系统环境描述	1
2.2.1	运行环境	1
2.2.2	开发环境	2
2.3	系统总体结构设计	3
2.3.1	系统业务层次图	3
2.3.2	模块功能介绍	3
3	包设计	4
3.1	整体架构设计	4
3.2	包的设计说明	5
4	类设计	5
5	接口设计	10
5.1	注册登录模块	10
5.1.1	模块内部接口	10
5.1.2	内部接口设计	10
5.2	搜索模块	10
5.2.1	模块内部接口	10
5.2.2	内部接口设计	10
5.3	展示模块	13
5.3.1	模块内部接口	13
5.3.2	内部接口设计	13
5.4	记录查询模块	15
5.4.1	模块内部接口	15
5.4.2	内部接口设计	15
5.5	支付模块	15
5.5.1	模块内部接口	15
5.5.2	内部接口设计	15

- 5.6 监控预警模块 15
 - 5.6.1 模块内部接口 15
 - 5.6.2 内部接口设计 15
- 5.7 排行榜功能 18
 - 5.7.1 模块内部接口 18
 - 5.7.2 内部接口设计 18
- 5.8 分析模块 18
 - 5.8.1 模块内部接口 18
 - 5.8.2 内部接口设计 18
- 6 数据库设计 24
 - 6.1 大数据分析模块 24
 - 6.2 数据存储系统的设计与实现 24

1 引言

1.1 编写目的

本概要说明书的编写目的在于明确说明该系统各功能的实现方法，以指导开发人员进行编码。

本文档的预期读者：系统设计人员、系统开发人员

1.2 背景

软件系统名称：影视舆情分析系统

软件系统任务提出者：汤新宇小组

软件系统任务开发者：汤新宇小组

2 总体设计

2.1 概述

本系统为基于大数据的舆情分析与预警系统，当今社会，互联网蓬勃发展，我们正处于一个一切皆有可能的大变革时代，纸媒、微博、微信、APP正在随时随地地影响着人们的生活，舆情场也随之改变，社会化媒体尤其是微博成为舆情爆发的主要阵地。本系统通过收集这些社会化媒体的数据，对当前比较热门的话题等进行舆情分析，并对该舆情的发展方向进行预测，对可能出现的负面影响预警。

本系统设计为 B / S 架构，前端采用 HTML 5 + CSS 3 实现，后端采用 Spring + Spring MVC +Hadoop 框架实现，本设计使系统具有优秀的解耦性，并大大增强了系统的可扩展性和可维护性。

2.2 系统环境描述

2.2.1 运行环境

- 软件环境

表 1: 软件环境

分类	名称	版本
操作系统	Linux(CentOS)	CentOS 7
数据库平台	hadoop	2.10.0
数据库平台	MongoDB	4.2
浏览器	IE	IE9及以上

- 硬件环境

表 2: 硬件环境

应用及服务器	最低配置	推荐配置
Mem	8G	32G
HD	160G	600G

2.2.2 开发环境

- 开发机器软件环境

表 3: 开发机器软件环境

分类	名称	版本
操作系统	Linux(CentOS)	CentOS 7
数据库平台	hadoop	2.10.0
数据库平台	MongoDB	4.2
浏览器	IE	IE9及以上

- 开发机器硬件环境

表 4: 开发机器硬件环境

应用及服务器	最低配置	推荐配置
Mem	8G	32G
HD	160G	600G

2.3 系统总体结构设计

2.3.1 系统业务层次图

影视舆情分析系统是针对影视舆情数据进行分析，为个人用户提供实时的和可定制的舆情事件展示和分析，为企业用户（诸如电视台、影视剧投资方等）提供舆情事件预警和营销效果分析的系统，具有实时性、准确定、个性化等特点，系统共由8个模块组成：

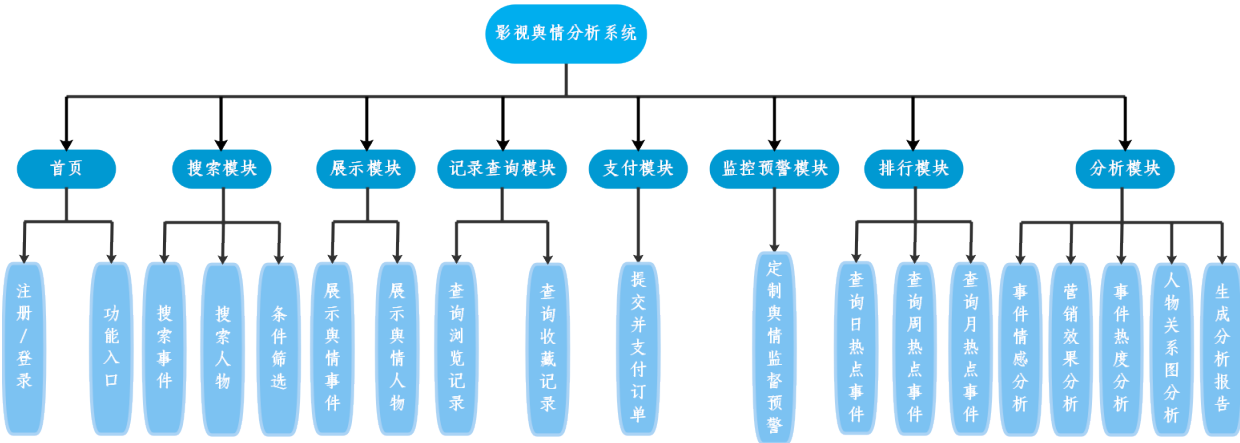


图 1: 系统业务层次图

2.3.2 模块功能介绍

- **首页：**提供用户注册、登录的功能按钮和个人/企业用户进入功能页面的模块；
- **搜索模块：**提供用户按照关键词搜索人物、时间的搜索功能或按一定条件（如职业、属地等）进行筛选的模块；
- **展示模块：**对事件发生的关键时间点，情感、热度变化和与人物有关的时间或关联人物可视化展示的模块；
- **记录查询模块：**提供对历史浏览记录和收藏查询的模块；
- **支付模块：**提供支付接口或查询支付记录的模块；
- **监控预警模块：**提供监控和预警定制和查看舆情预警的模块；
- **排行模块：**提供不同周期的热点事件排行榜的模块，具体时间周期为单日、本周和本月的热点事件排行榜；

- **分析模块：**提供影视舆情数据分析模块，可以对事件情感、热度和人物关系的分析进行查询，而企业用户可以对营销效果进行分析并生成分析报告。

3 包设计

3.1 整体架构设计

本系统为了能更加有效地进行整合、生产宏观模型，就需要对系统中的类进行分组，以下是本系统中的包设计。

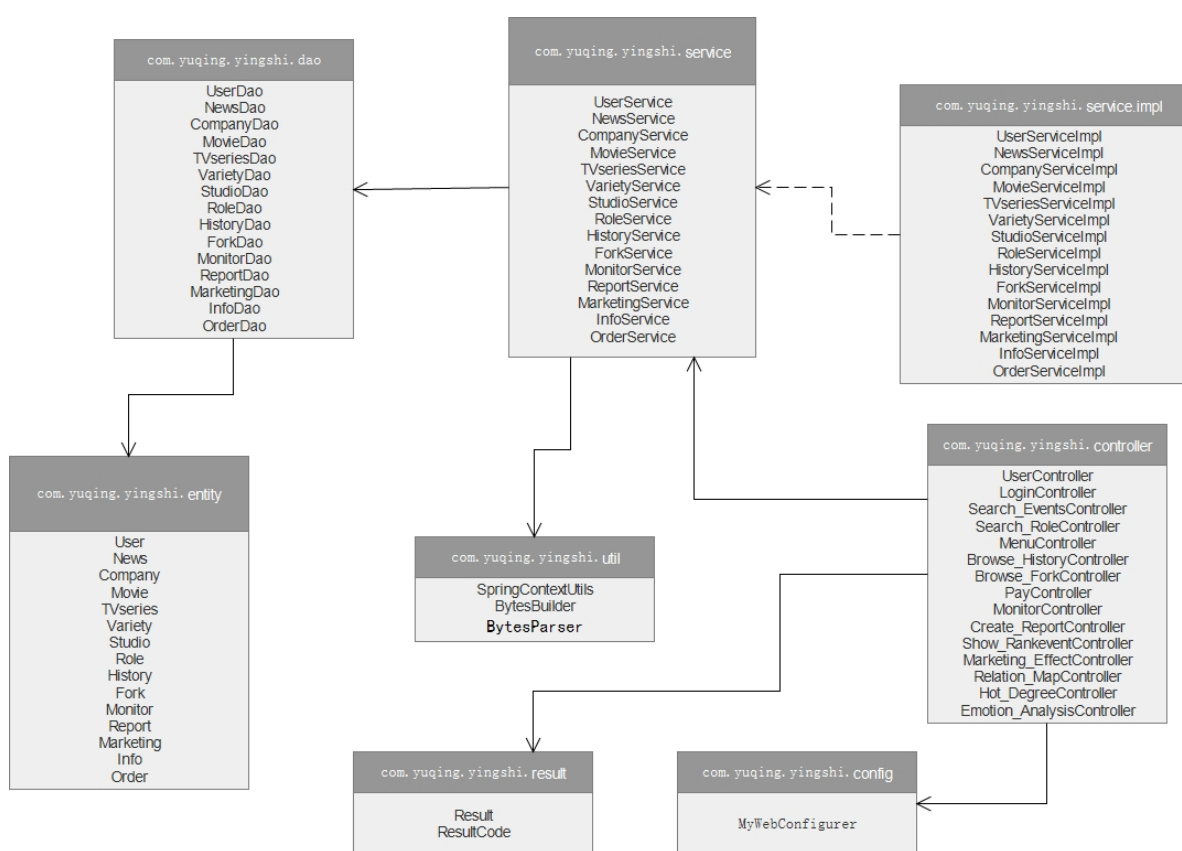


图 2: 舆情系统包的设计图

3.2 包的设计说明

下表为服务器的包设计表

表 5: 包设计表

包名	设计说明
com.yuqing.yingshi.service	该包主要存放了高层调用Dao的接口
com.yuqing.yingshi.controller	该包主要存放了业务逻辑相关类，依赖于com.yuqing.yingshi.service，调用Service以完成业务逻辑
com.yuqing.yingshi.result	该包存放了服务器与web端交互时返回的结果对应编码
com.yuqing.yingshi.dao	主要负责数据访问，该包中的类封装了对数据库的访问（只包含最原子的数据操作），供高层调用，依赖于com.yuqing.yingshi.entity
com.yuqing.yingshi.config	存放了与前端web网页交互的相关配置
com.yuqing.yingshi.entity	该包主要存放数据库表中所对应的实体类
com.yuqing.yingshi.service.impl	该包存放了service包中定义的接口的具体实现
com.yuqing.yingshi.util	存放了项目相关工具类

4 类设计

本小结以包为单位，对每个包所含有的类及类与类间的调用关系进行说明。

表 6: com.yuqing.yingshi.dao相关类

com.yuqing.yingshi.dao相关类	
UserDao	封装与User实体类相关的数据访问方法
NewsDao	封装与News实体类相关的数据访问方法
CompanyDao	封装与Company实体类相关的数据访问方法
MovieDao	封装与Movie实体类相关的数据访问方法
TVseriesDao	封装与TVseries实体类相关的数据访问方法
VarietyDao	封装与Variety实体类相关的数据访问方法
StudioDao	封装与Studio实体类相关的数据访问方法
RoleDao	封装与Role实体类相关的数据访问方法
HistoryDao	封装与History实体类相关的数据访问方法
ForkDao	封装与Fork实体类相关的数据访问方法
MonitorDao	封装与Monitor实体类相关的数据访问方法
ReportDao	封装与Report实体类相关的数据访问方法
MarketingDao	封装与Marketing实体类相关的数据访问方法
InfoDao	封装与Info实体类相关的数据访问方法
OrderDao	封装与Order实体类相关的数据访问方法

表 7: com.yuqing.yingshi.entity相关类

com.yuqing.yingshi.entity相关类	
User	用户个人信息的实体类
News	新闻信息的实体类
Company	公司信息的实体类
Movie	电影信息的实体类
TVseries	影视剧信息的实体类
Variety	综艺信息的实体类
Studio	工作室信息的实体类
Role	演员/歌手等明星信息的实体类
History	历史记录信息的实体类
Fork	收藏记录的实体类
Monitor	监控预警项的实体类
Report	报告的实体类
Marketing	营销效果的实体类
Info	单条博文/帖子信息的实体类
Order	用户订单的实体类

表 8: com.yuqing.yingshi.service相关类

com.yuqing.yingshi.service相关类	
UserService	定义了与User实体类相关的业务逻辑接口
NewsService	定义了与News实体类相关的业务逻辑接口
CompanyService	定义了与Company实体类相关的业务逻辑接口
MovieService	定义了与Movie实体类相关的业务逻辑接口
TVseriesService	定义了与TVseries实体类相关的业务逻辑接口
VarietyService	定义了与Variety实体类相关的业务逻辑接口
StudioService	定义了与Studio实体类相关的业务逻辑接口
RoleService	定义了与Role实体类相关的业务逻辑接口
HistoryService	定义了与History实体类相关的业务逻辑接口
ForkService	定义了与Fork实体类相关的业务逻辑接口
MonitorService	定义了与Monitor实体类相关的业务逻辑接口
ReportService	定义了与Report实体类相关的业务逻辑接口
MarketingService	定义了与Marketing实体类相关的业务逻辑接口
InfoService	定义了与Info实体类相关的业务逻辑接口
OrderService	定义了与Order实体类相关的业务逻辑接口

表 9: com.yuqing.yingshi.util相关类

com.yuqing.yingshi.util相关类	
SpringContextUtils	获取spring容器bean对象工具类
BytesBuilder	字节构造器
BytesParser	字节处理器

表 10: com.yuqing.yingshi.result相关类

com.yuqing.yingshi.result相关类	
Result	结果返回形式定义
ResultCode	定义返回结果编码

表 11: com.yuqing.yingshi. controller相关类

com.yuqing.yingshi. controller相关类	
UserControlle	定义了响应用户相关界面不同点击事件的方法，根据事件的不同调用 UserService的相应方法实现功能，以json格式返回。
LoginController	定义了登陆注册界面不同点击事件的方法，根据事件的不同调用 UserService的相应方法实现功能，以json格式返回。
Search_EventsController	定义了搜索热门影视事件/影视作品相关界面不同点击事件的方法，根据事件的不同调用 NewsService、MovieService、TVseriesService、VarietyService的相应方法实现功能，以json格式返回。
Search_RoleController	定义了搜索明星/公司/工作室相关界面不同点击事件的方法，根据事件的不同调用 Role-Service、CompanyService、StudioService的相应方法实现功能，以json格式返回。
MenuController	定义了展示影视事件/明星/影视作品菜单相关界面不同点击事件的方法，根据事件的不同调用 RoleService、CompanyService、StudioService、NewsService、MovieService、TVseriesService、VarietyService的相应方法实现功能，以json格式返回。
Browse_HistoryController	定义了浏览历史记录相关界面不同点击事件的方法，根据事件的不同调用 HistoryService的相应方法实现功能，以json格式返回。
Browse_ForkController	定义了浏览收藏记录相关界面不同点击事件的方法，根据事件的不同调用 ForkService的相应方法实现功能，以json格式返回。

表 12: com.yuqing.yingshi. controller相关类 (续)

com.yuqing.yingshi. controller相关类	
PayController	定义了会员支付相关界面不同点击事件的方法，根据事件的不同调用 PayService的相应方法实现功能，以json格式返回。
MonitorController	定义了监控预警相关界面不同点击事件的方法，根据事件的不同调用 MonitorService的相应方法实现功能，以json格式返回。
Create_ReportController	定义了生成报告相关界面不同点击事件的方法，根据事件的不同调用 ReportService的相应方法实现功能，以json格式返回。
Show_RankeventController	定义了热门事件排行榜相关界面不同点击事件的方法，根据事件的不同调用 NewsService的相应方法实现功能，以json格式返回。
Marketing_EffectController	定义了营销效果分析与追踪相关界面不同点击事件的方法，根据事件的不同调用 NewsService的相应方法实现功能，以json格式返回。
Relation_MapController	定义了明星关系图谱相关界面不同点击事件的方法，根据事件的不同调用 RoleService的相应方法实现功能，以json格式返回。
Hot_DegreeController	定义了分析事件热度相关的功能与方法，根据事件的不同调用 RoleService、CompanyService、StudioService、NewsService、MovieService、TVseriesService、VarietyService的相应方法实现功能，以json格式返回。
Emotion_AnalysisController	定义了情感分析相关的功能与方法，根据事件的不同调用RoleService、CompanyService、StudioService、NewsService、MovieService、TVseriesService、VarietyService的相应方法实现功能，以json格式返回。

表 13: com.yuqing.yingshi.config相关类

com.yuqing.yingshi.config相关类	
MyWebConfigurer	定义了与前端web页面交互的相关配置

5 接口设计

5.1 注册登录模块

5.1.1 模块内部接口

外部接口	内部调用函数	函数说明
User_SignUp	addUser	用户注册，将用户的输入信息插入数据库用户表中。
User_SignIn	getUserById	用户登录，在用户表中比对用户输入的用户名和密码是否存在并匹配，返回值决定弹窗提示信息。

图 3: 注册登录模块内部接口

5.1.2 内部接口设计

- 注册实现接口
- 登录实现接口

5.2 搜索模块

5.2.1 模块内部接口

5.2.2 内部接口设计

- 搜索热点事件
- 搜索热点明星

函数名	addUser		
功能概要	用户注册，将用户输入信息插入数据库用户表中。		
记述形式	Boolean addUser(User user)		
参数			
类型	变量名	I/O	说明
User	user	用户输入	无
返回值	类型	Boolean	
	说明	插入成功返回 True，否则返回 False。	

图 4: 注册实现接口

函数名	getUserById		
功能概要	用户登录，在用户表中比对用户输入的用户名和密码是否存在并匹配。		
记述形式	Boolean getUserById (User user)		
参数			
类型	变量名	I/O	说明
User	user	用户输入	参数实际为输入 用户名及密码
返回值	类型	Boolean	
	说明	匹配成功返回 True， 否则返回 False。	

图 5: 登录实现接口

外部接口	内部调用函数	函数说明
User_Search	searchEvents	用户搜索自己感兴趣的事件，用户通过输入关键字来搜索特定事件。
	searchRole	用户搜索自己感兴趣的明星，用户通过输入姓名来搜索特定明星。

图 6: 搜索模块内部接口

函数名	searchEvents		
功能概要	用户通过输入关键字来搜索特定事件。		
记述形式	List<News> searchEvents(String info)		
参数			
类型	变量名	I/O	说明
String	info	用户输入	输入事件关键字
返回值	类型	List<News>	
	说明	返回热点事件的相关信息。	

图 7: 搜索热点事件实现接口

函数名	searchRole		
功能概要	用户通过输入姓名来搜索特定明星。		
记述形式	Role searchRole(String name)		
参数			
类型	变量名	I/O	说明
String	name	用户输入	输入姓名
返回值	类型	Role	
	说明	返回人物的相关信息。	

图 8: 搜索热点明星实现接口

5.3 展示模块

5.3.1 模块内部接口

外部接口	内部调用函数	函数说明
Menu_Show	showEvents	首页搜索栏下面为当今热点事件的展示。
	showRoles	首页搜索栏下面为当今热点明星相关信息的展示。

图 9: 展示模块内部接口

5.3.2 内部接口设计

- 展示热点事件

函数名	showEvents		
功能概要	在搜索栏下面展示当今热点事件。		
记述形式	List<News> showEvents()		
参数			
类型	变量名	I/O	说明
无	无	无	无
返回值	类型	List<News>	
	说明	返回热点事件的相关信息。	

图 10: 展示热点事件实现接口

● 展示热点明星

函数名	showRoles		
功能概要	在搜索栏下面展示当今热点明星。		
记述形式	List<Roles> showRoles()		
参数			
类型	变量名	I/O	说明
无	无	无	无
返回值	类型	List<Roles>	
	说明	返回热点明星的相关信息。	

图 11: 展示热点明星实现接口实现

5.4 记录查询模块

5.4.1 模块内部接口

外部接口	内部调用函数	函数说明
Browse_Record _Query	browseHistoryQuery	用户查询自己的浏览记录并且展示在界面上。
	browseForkQuery	用户查询自己的收藏记录并且展示在界面上。

图 12: 记录查询模块内部接口实现

5.4.2 内部接口设计

- 查询浏览记录
- 查询收藏记录

5.5 支付模块

5.5.1 模块内部接口

5.5.2 内部接口设计

- 支付功能

5.6 监控预警模块

5.6.1 模块内部接口

5.6.2 内部接口设计

- 监控预警功能

函数名	browseHistoryQuery		
功能概要	用户查询自己的浏览记录并且展示在界面上。		
记述形式	List<History> browseHistoryQuery (User user)		
参数			
类型	变量名	I/O	说明
User	user	无	传入用户相关信息
返回值	类型	List<History>	
	说明	返回该用户的浏览记录。	

图 13: 查询浏览记录接口实现

函数名	browseForkQuery		
功能概要	用户查询自己的浏览记录并且展示在界面上。		
记述形式	List<Fork> browseForkQuery (User user)		
参数			
类型	变量名	I/O	说明
User	user	无	传入用户相关信息
返回值	类型	List<Fork>	
	说明	返回该用户的收藏记录。	

图 14: 查询收藏记录接口实现

外部接口	内部调用函数	函数说明
Pay	sumMoney	用户提交订单后进行支付。

图 15: 支付模块内部接口

函数名	sumMoney		
功能概要	用户提交订单后进行支付。		
记述形式	Boolean sumMoney (Order order)		
参数			
类型	变量名	I/O	说明
Order	order	无	传入用户提交订单的相关信息
返回值	类型	Boolean	
	说明	返回 True 代表支付成功, False 代表支付失败。	

图 16: 支付功能接口实现

外部接口	内部调用函数	函数说明
Monitor	monitorWarning	对某特定时间进行监督预警。

图 17: 监控预警模块内部接口

函数名	monitorWarning		
功能概要	对某特定时间进行监督预警。		
记述形式	Monitor MonitorWarining(News news)		
参数			
类型	变量名	I/O	说明
News	news	无	传入特定事件的 相关信息
返回值	类型	Monitor	
	说明	返回监控预警项。	

图 18: 监控预警功能实现

5.7 排行榜功能

5.7.1 模块内部接口

5.7.2 内部接口设计

- 查询日排名
- 查询周排名
- 查询月排名

5.8 分析模块

5.8.1 模块内部接口

5.8.2 内部接口设计

- 生成分析报告功能

外部接口	内部调用函数	函数说明
Rank_Query	rankByDay	用户查询热点事件日排名并且显示在界面上。
	rankByWeek	用户查询热点事件周排名并且显示在界面上。
	rankByMonth	用户查询热点事件月排名并且显示在界面上

图 19: 排行榜模块内部接口

函数名	rankByDay		
功能概要	用户查询热点事件日排名。		
记述形式	List<News> rankByDay()		
参数			
类型	变量名	I/O	说明
无	无	无	无
返回值	类型	List<News>	
	说明	返回热点事件日排名及相关信息。	

图 20: 查询日排名接口实现

函数名	rankByWeek		
功能概要	用户查询热点事件周排名。		
记述形式	List<News> rankByWeek()		
参数			
类型	变量名	I/O	说明
无	无	无	无
返回值	类型	List<News>	
	说明	返回热点事件周排名及相关信息。	

图 21: 查询周排名接口实现

函数名	rankByMonth		
功能概要	用户查询热点事件月排名。		
记述形式	List<News> RankByMonth()		
参数			
类型	变量名	I/O	说明
无	无	无	无
返回值	类型	List<News>	
	说明	返回热点事件月排名及相关信息。	

图 22: 查询月排名接口实现

外部接口	内部调用函数	函数说明
Generate_Analysis_Report	generateAnalysisReport	根据特定的事件系统自动生成舆情分析报告。
Marketing_Effect_Analysis	marketingEffectAnalysis	根据特定的事件进行营销效果分析。
HotDegree_Analysis	hotDegreeAnalysis	对特定的事件进行热度分析。
Emotion_Analysiss	emotionAnalysis	对特定的事件进行情感分析。
Generate_RelationMap	generateRelationMap	对特定的人物生成人物关系图谱。

图 23: 分析模块内部接口

- 营销效果分析功能
- 热度分析功能
- 情感分析功能
- 人物关系图谱功能

函数名	generateAnalysisReport		
功能概要	根据特定的事件系统自动生成舆情分析报告。		
记述形式	Report generateAnalysisReport (News news)		
参数			
类型	变量名	I/O	说明
News	news	无	特定热点事件
返回值	类型	Report	
	说明	返回特定事件的舆情分析报告。	

图 24: 生成分析报告接口实现

函数名	marketingEffectAnalysis		
功能概要	根据特定的事件进行营销效果分析。		
记述形式	Marketing marketingEffectAnalysis (News news)		
参数			
类型	变量名	I/O	说明
News	news	无	特定热点事件
返回值	类型	Marketing	
	说明	返回特定事件的营销效果。	

图 25: 营销效果分析接口实现

函数名	hotDegreeAnalysis		
功能概要	对特定的事件进行热度分析。		
记述形式	Report hotDegreeAnalysis (News news)		
参数			
类型	变量名	I/O	说明
News	news	无	特定热点事件
返回值	类型	Report	
	说明	返回特定事件的热度分析报告。	

图 26: 热度分析接口实现

函数名	emotionAnalysis		
功能概要	对特定的事件进行情感分析。		
记述形式	Report emotionAnalysis (News news)		
参数			
类型	变量名	I/O	说明
News	news	无	特定热点事件
返回值	类型	Report	
	说明	返回特定事件的情感分析报告。	

图 27: 情感分析接口实现

函数名	generateRelationMap		
功能概要	对特定的人物生成人物关系图谱。		
记述形式	Gragh generateRelationMap (Role role)		
参数			
类型	变量名	I/O	说明
Role	role	无	特定人物
返回值	类型	Gragh	
	说明	返回特定人物的人物关系图谱。	

图 28: 生成人物关系图谱接口实现

6 数据库设计

6.1 大数据分析模块

大数据分析模块负责进行数据的处理，它的整体框架图是这样的:

6.2 数据存储系统的设计与实现

介绍 数据存储系统是大数据可视化的基础，是整个系统的根脉。对于存储的大量数据进行统计，计算和处理通常需要以小时和天为记的时间，而本系统的用户会根据需要对于已经处理好的数据进行可视化定制。所以数据存储系统的设计需要满足以下用户需求：

- 对已处理的数据进行可视化渲染
- 对结构简单的数据进行多样化的渲染，比如同时渲染成柱状图和饼图
- 对渲染的低延时需求，需要“即做即画”
- 对某组数据进行快速索引，通过少量关键字迅速检索需要的数据

基于以上需求特点，本系统采用Hadoop生态中的HBase加Mysql数据库存储方案，该方案的基本流程是：将所有需要的数据，放入hdfs中，然后用hadoop进行分析，得出

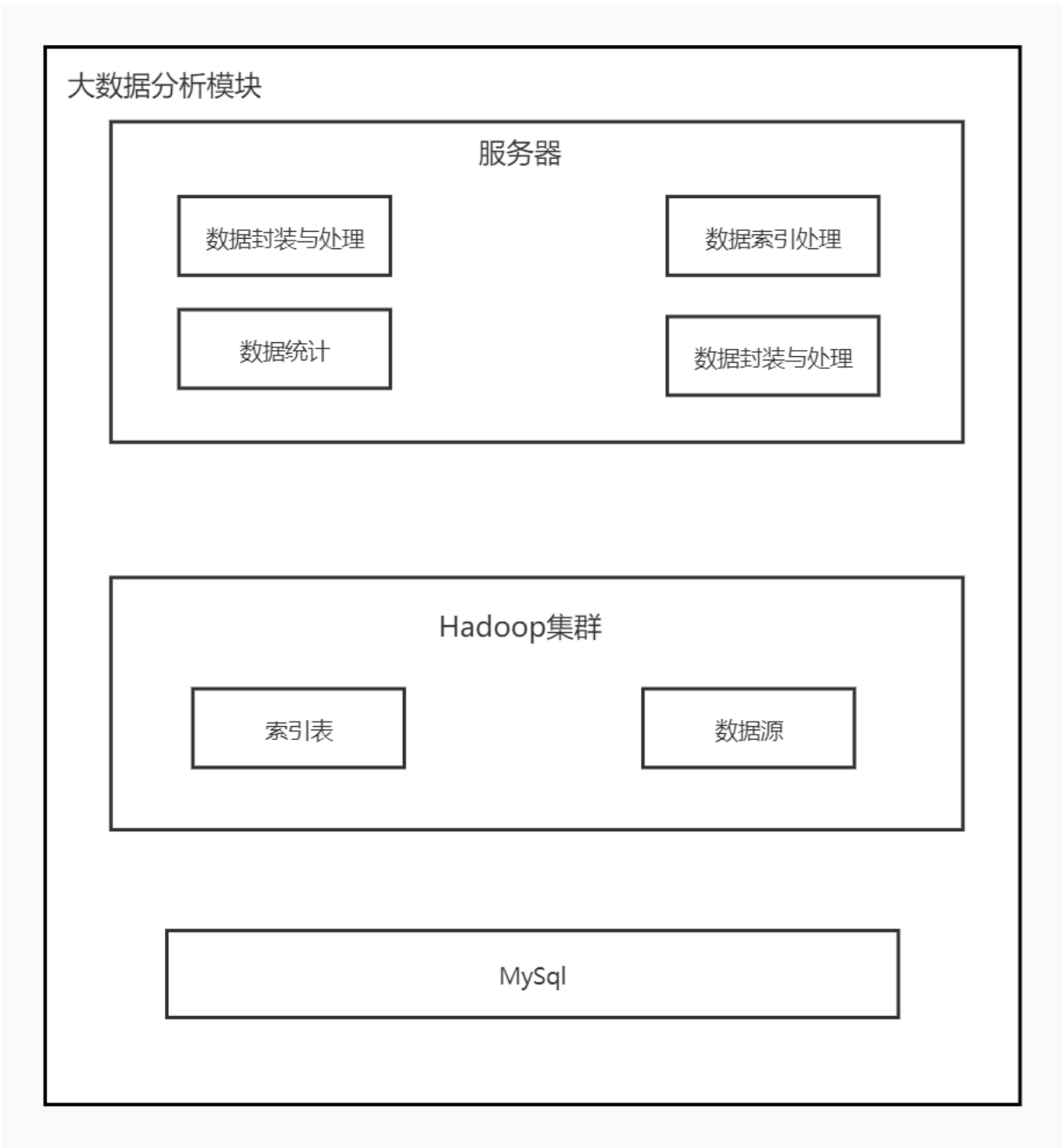


图 29: 大数据分析模块

来最终的结果数据，再放入mysql数据库表中。前端接口再查询访问这些最终的结果数据。具有以下特点：

- 擅长存储半结构化数据，存储结构灵活
- 面向列的设计使得一张表中可以存储数以万计的不同图表数据
- 可以提供低延时查询，通过行键的查询延时在1ms内

该方案的好处：

- 所有的数据放入了hdfs中，而不放入线上mysql数据库中，这样在统计分析数据时，就不会影响正在操作数据库的用户。
- 利用了hadoop的“分布”式“计算”框架的优势，可以像多核cpu一起共同计算一样，这样分析速度会快。
- 减少了线上mysql数据库的负担。
- 所有需要的数据，都放在hdfs中，这样让相关人员的思路也会觉得清晰。

HBase的基本存储方式

- 表名

表 14: 表名

行键	列族
通过键值检索行	在创建表时定义所有列族
通常存储全表最关键的索引信息	每行中每个列族可以存储任意多列
一个行键的大小最大为单元格大小	一行和一个列族和一个列对应一个单元格
行键在全表中唯一	可以重复，不同的单元格用时间戳区分
每一行的每个列族下的列都不一样	可以重复，不同的单元格用时间戳区分，查询时一般查询最新单元格