# 引言

## 选题背景

## 文献综述

## 研究目的及意义

## 论文主要工作

## 论文章节安排

# 相关技术背景

# 需求分析

## 功能需求分析

智能投研热点挖掘系统的目标客户为买方机构用户，包括公募基金、信托、银行理财子、券商自营、PE/VC等机构，同时也可以作为券商资讯增值服务提供给个人客户使用。

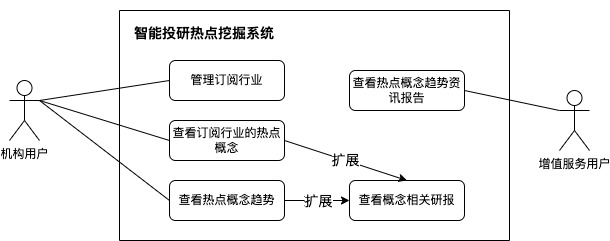
对市场热点概念的趋势把握是买方投研的重点，因此系统的核心功能是能够提供行业的热点概念发现和概念趋势的跟踪。

结合买方的特点，一般用户会重点关注某个行业或某几个行业，因此系统需要提供行业订阅的功能。

热点概念和趋势的显示并不能解决投研人员对具体信息想要掌握更多细节的问题，因此还需要提供详细的研报阅读的功能，让热点概念和研报内容相绑定。

结合上述分析针对买方机构用户需要实现的功能用例主要有：管理订阅行业、查看订阅行业的热点概念、查看热点概念趋势、查看概念相关研报。

对于资讯增值服务，只需要提供热点概念趋势的功能即可。



### 管理订阅行业

不同机构用户一般有自己所重点关注的行业，因此需要提供行业订阅的功能，行业订阅的功能会影响查看订阅行业热点概念用例的显示。

一个用户应能够订阅多个行业，行业的分类标准和研究报告发布的标准相同。

对没有设置订阅行业的用户应提示其进行行业的订阅，不设置默认的订阅行业。

### 查看订阅行业的热点概念

查看订阅行业的热点概念会显示用户已经订阅的行业内的热点概念。热点概念是能够反映一段时间内研究报告重点关注的话题。热点概念的显示应该能够体现不同概念的热度。

### 查看热点概念趋势

热点概念趋势显示所选范围内的概念热度变化情况，以排名的形式进行显示。默认显示全部的行业的概念趋势排行。可以选择到某一个行业的情况。

### 查看概念相关研报

用户在查看热点概念和概念趋势时，如果想深度了解某一概念的内容，可以通过点击概念名词的方式来查看概念相关研究报告。

为了给用户提供便捷的概念研究报告阅读体验，应提供列表形式的摘要阅读，需要阅读某篇研报的详细内容再通过点击列表选项等形式进入到原文进行阅读。

### 查看热点概念趋势资讯报告

热点概念趋势资讯报告作为面向个人用户提供的增值服务类产品，以资讯报告的形式交付给客户进行使用。

## 性能需求分析

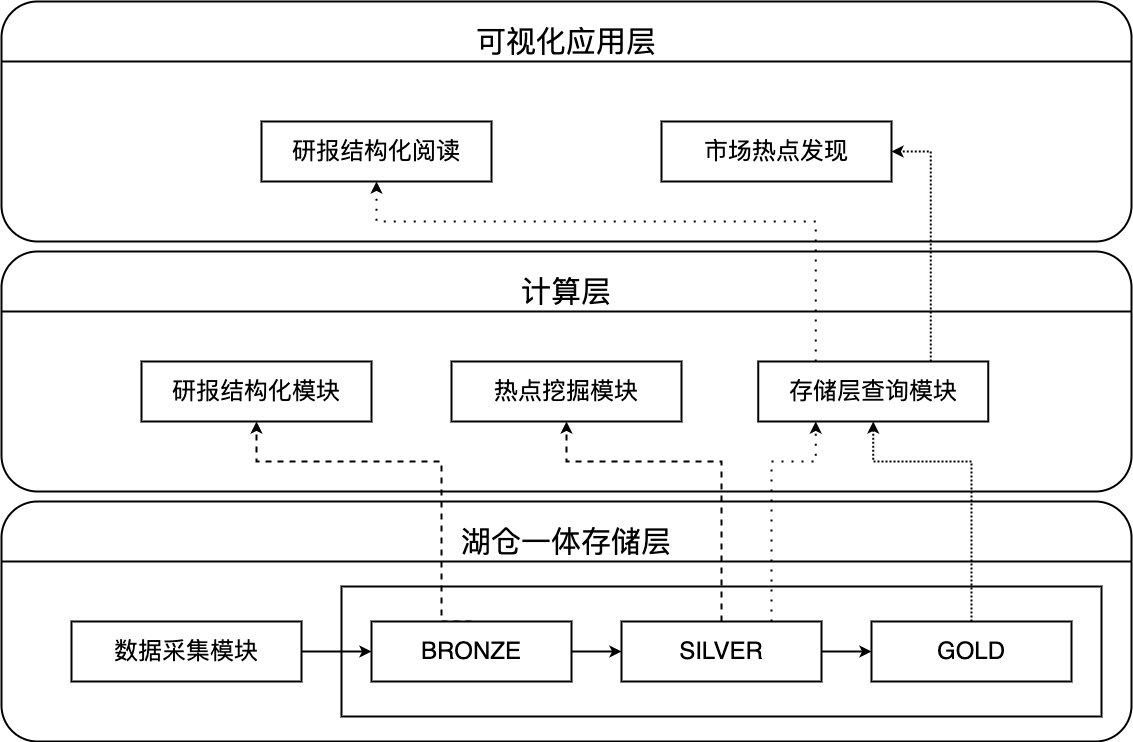
系统以SAAS化的方式对客户提供服务以最大化经济效应。面向机构客户的服务内容应能够应对1000人的并发访问。响应时间应该控制在3秒以内。

研报数据的原始形态一般为PDF格式的文件，其中包含大量的图片。平均每篇研报的大小在10MB左右，根据每日500篇研报测算，每日新增存储在5GB。系统应能够存储近前后10年内的所有研报，存储需求大约为40TB。

# 系统设计

在大数据架构的搭建中，Lamda架构一直是主流方案，对批处理和实时计算采用两套计算方案。这样的方案提高了大数据开发和维护的门槛，同时耗费了大量的人力物力。Lakehouse湖仓一体概念的提出，简化了Lamda架构的搭建。DeltaLake是一个开源的面向湖仓一体架构的存储层引擎，其具备的特点：ACID事务、可伸缩的元数据处理、时间回溯（数据版本控制）、统一批处理和流源和接收器。使用DeltaLake架构可以替代当前繁琐的Lamda架构，简化大数据平台的数据流管理，构建流批一体的数据湖架构。DeltaLake可以使用HDFS、Data Lake、S3等作为底层存储，同时上层计算接口完全兼容SPARK原有接口。

### 整体架构设计



系统整体分为三层，从数据进入系统到应用分为湖仓一体存储层、计算层、可视化应用层。

湖仓一体存储层负责整个大数据架构的存储，为计算层提供分析模型所需要的数据准备。湖仓一体存储层包括两大部分，意识数据采集模块，从各种研报数据源采集研报数据进入系统。数据采集模块采集的研报数据会被写入使用湖仓一体存储引擎DELTALAKE控制的对象存储中。存储采用湖仓一体的三层经典架构：

Bronze表：存储进入系统的原生数据，获取到研报数据不做修改的存储。

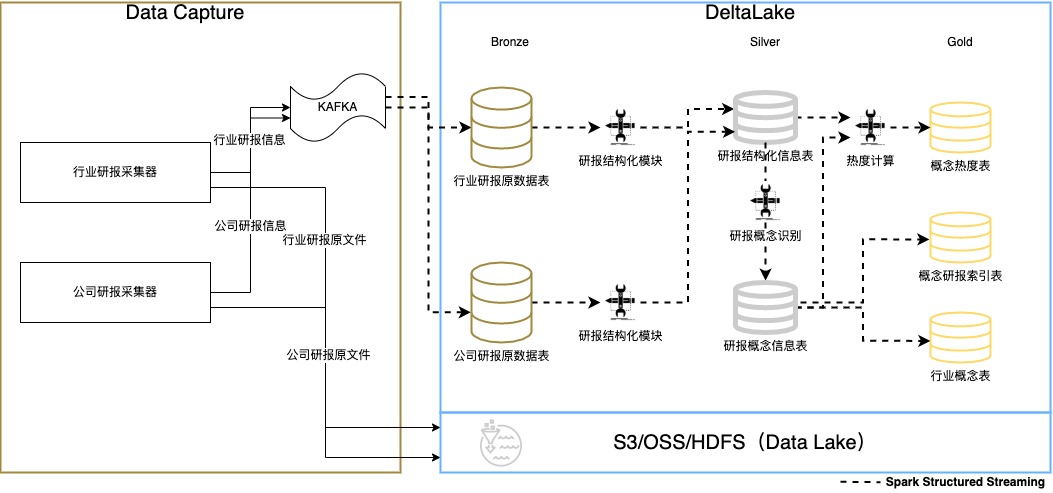
Silver表：该表是在 Bronze表的基础上进行加工的到的中间数据，一般对Bronze中的原始数据进行转换清洗，成为数据科学训练的数据源。

Gold表：该表是基于业务需求的数据表，表数据是高度集成的，用于最上层的可视化应用。

计算层提供了系统中所有计算相关的功能，包括了湖仓一体层数据转换的功能模块，研报进入原生数据表即Bronze表后会使用研报结构化模块将文件形式的研报转换成结构形式的研报送入Silver层的表中。进入Silver层的表的数据会被热点挖掘模块作为数据源来进行分析，得到热点趋势分析结果数据，分析的结果数据会被送入Gold层的表中。存储层的查询模块因为涉及到多个存储层数据的查询和查询策略算法的设计所以也放在计算层中。存储查询模块从Silver层的表中查询结构化处理后的研报数据提供给可视化应用层中的研报结构化模块，从Gold层的表中查询热点分析后的结果数据提供给可视化应用层中的市场热点发现模块，从Bronze层中查询研报原始数据供用户预览下载。

最上层是面向用户的可视化应用层，这一层的内容不是固定不变的，随着用户应用场景的扩充会不断有新的应用场景诞生，就需要在应用层实现更多的模块。当需要实现新的功能时计算层需要增加对应的模块提供数据，向湖仓一体存储层传导就会新增一些对应的表内容。

### 存储层设计



存储层在整体上分为数据采集模块和湖仓一体存储模块两大部分。

如图，左边为数据采集模块DC（Data Capture）部分。根据对源数据的分析可以用两个研报采集器进行覆盖，分别为行业研报采集器和公司研报采集器。采集器从数据源获取研究报告原文，和研报相关信息（作者、发布时间、机构、所属行业等）。将研究报告的原文存储到数据湖中，考虑到研究报告原文数据量很大，架构中选用经济性突出的公有云对象存储；将研报相关信息和原文存储路径存储到DeltaLake的Bronze层中。

如图，右边为湖仓一体的存储模块。DeltaLake同样使用研报原文存储的对象存储作为存储单元，使得整体架构非常精简，维护成本大幅降低。虚线表示使用Spark Structured Streaming产生的数据流。研报相关信息被送入KAFKA后，就成为第一个数据流的写入并且输出到湖仓一体存储模块的Bronze层中，该层不对数据做任何处理，存储原始文件和原始信息。随后进入Bronze层中的数据成为下一个数据流的写入，通过研报数据化模块的转换，将研报原文和相关信息转换成结构化的数据存储到Silver层的表中供热点挖掘模块使用，热点挖掘模块使用Silver层中的结构化数据得出热点分析结果数据写入到Gold层中的热点分析结果表。

#### 研报采集模块设计

获取研报的数据源是从API形式的订阅接口获取，包括了行业研究报告和公司研究报告。

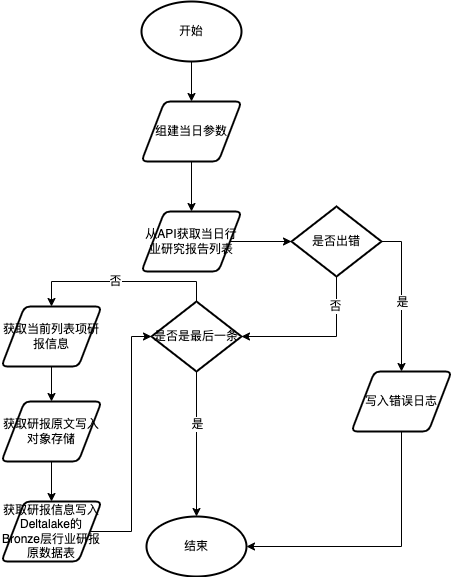
行业研究报告的发布往往没有特定时间点，一般聚焦在某一段时间的行业热点问题，但是为了实现行业热点挖掘功能，必须至少满足日频的采集频率。

公司研究报告的发布，一类是和行业研究报告发布具备同样特性的不定期覆盖报告，另一类是事件驱动的点评类报告，其中以上市公司公告事件为主要。

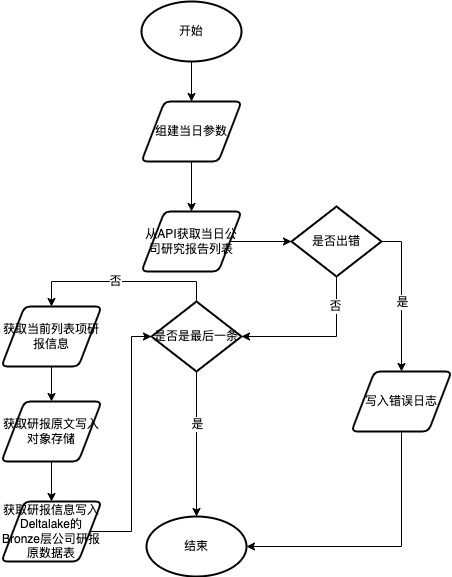
根据上述对数据源的分析，需要实现三类数据采集器。

1. API接口行业研究报告采集器（日频采集，覆盖所有研究领域）
2. API接口公司研究报告采集器（日频采集，覆盖所有上市公司）

API接口行业研究报告采集器的采集过程：



API接口公司研究报告采集器的采集过程：

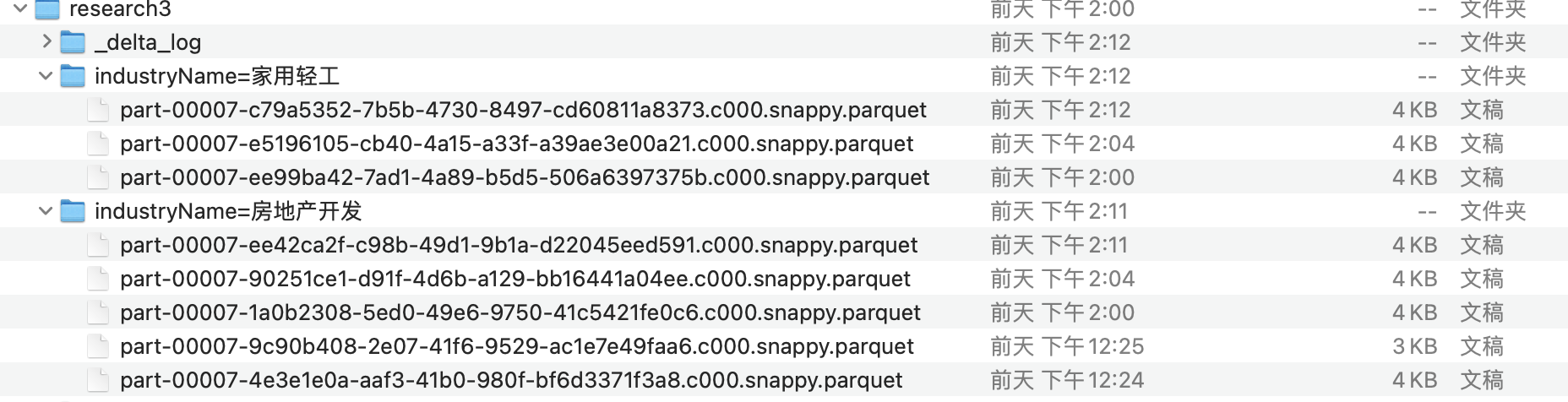


两个API接口的采集器逻辑上基本相同，都是先组建相应的API参数获取到研究报告的列表信息，然后遍历这个列表去获取详细的内容。获取详细内容后，首先将研报原文写入到数据湖中也就是云上的对象存储中去，确定相关研究报告存储到对象存储中后将包括对象存储位置相关信息存储到DeltaLake的Bronce层对应的表中。

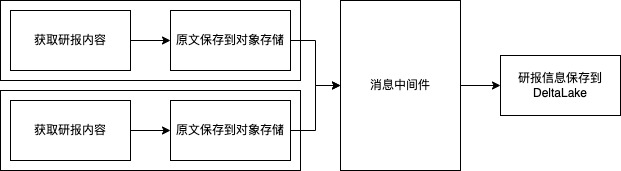
处理的整体过程如下：



但是这样的过程会产生Deltalake存储中小Parquet文件过多的问题：



每次存储一个研究报告都会在对应行业目录下产生一个小的parquet文件，在大量小parquet文件增多的情况下，查询相关行业内的内容操作速度会产生严重影响。为了解决这个问题，有两个解决方案，一是在写入的Bronce层的时候进行批量的写入，这就需要有一个地方来进行缓存，无疑消息中间件是最佳选择。加入消息中间件后形成的处理流程如下：



研报采集模块作为生产者不断将研报信息存储到消息中间件中，将研报信息保存到DeltaLake的消费者一次性读取大量的研报信息内容存储到Bronce层。这这样可以避免每次写入一条研报信息都产生一个小的parquet文件。

为了避免特殊情况，比如某日某行业相关的研报非常少，也将产生大量的小parquet文件，为了解决这样的问题，需要设计相应的处理模块定期的将小的parquet文件合并成大小合适的parquet文件。

#### 研报数据存储模块设计

**存储**

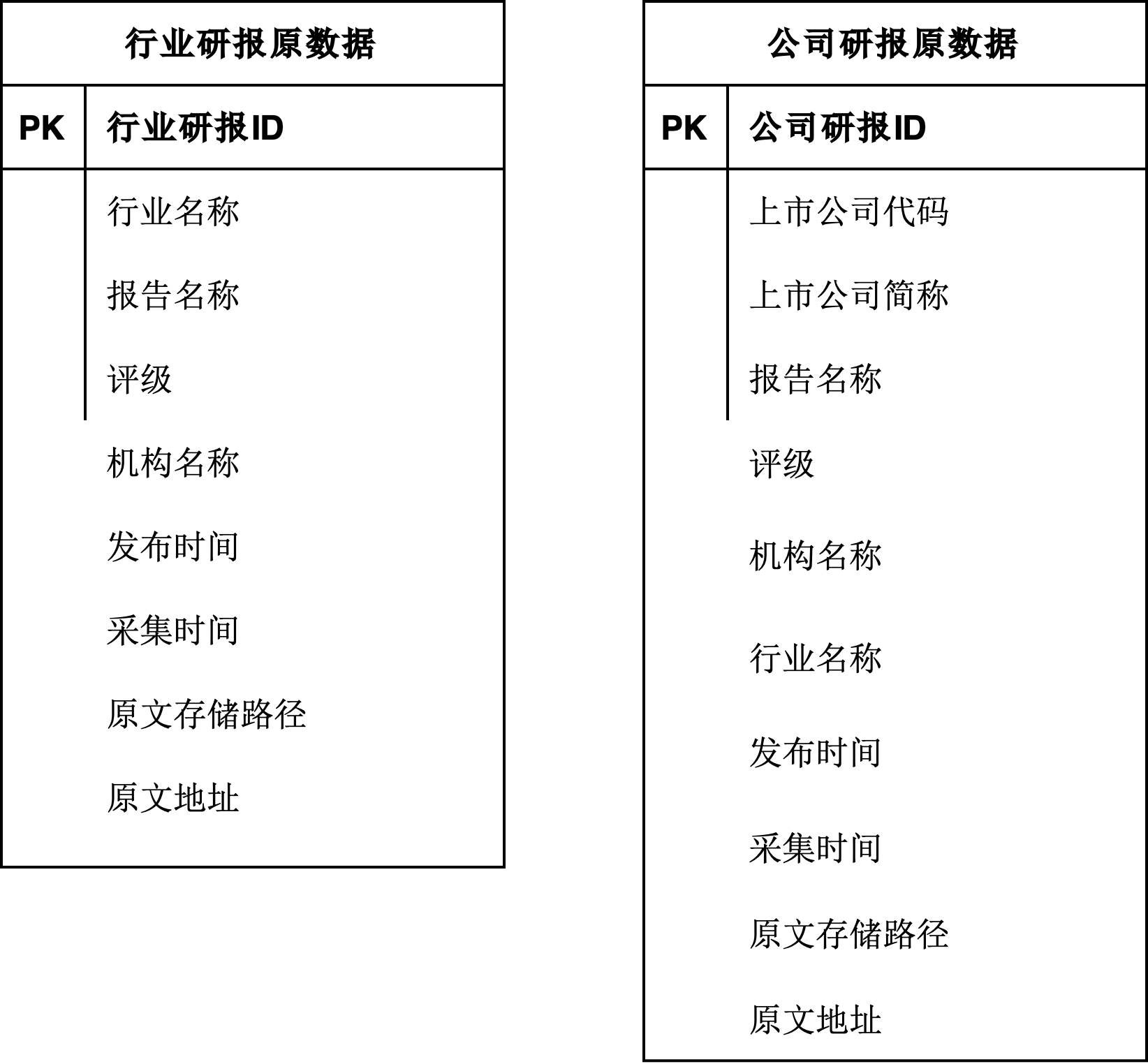
存储模块采用DeltaLake为存储引擎，底层存储采用对象存储，可以选用HDFS、AWS S3、阿里云OSS等支持DeltaLake接口的对象存储。

使用DeltaLake的最直接好处是数据流转换的三层中的表都以PARQUET格式存储在对象存储上。研报原文件也同样存储在对象存储之上。架构较过往采用的LAMDA和KAPPA架构有了极大的精简。

**三层数据架构**

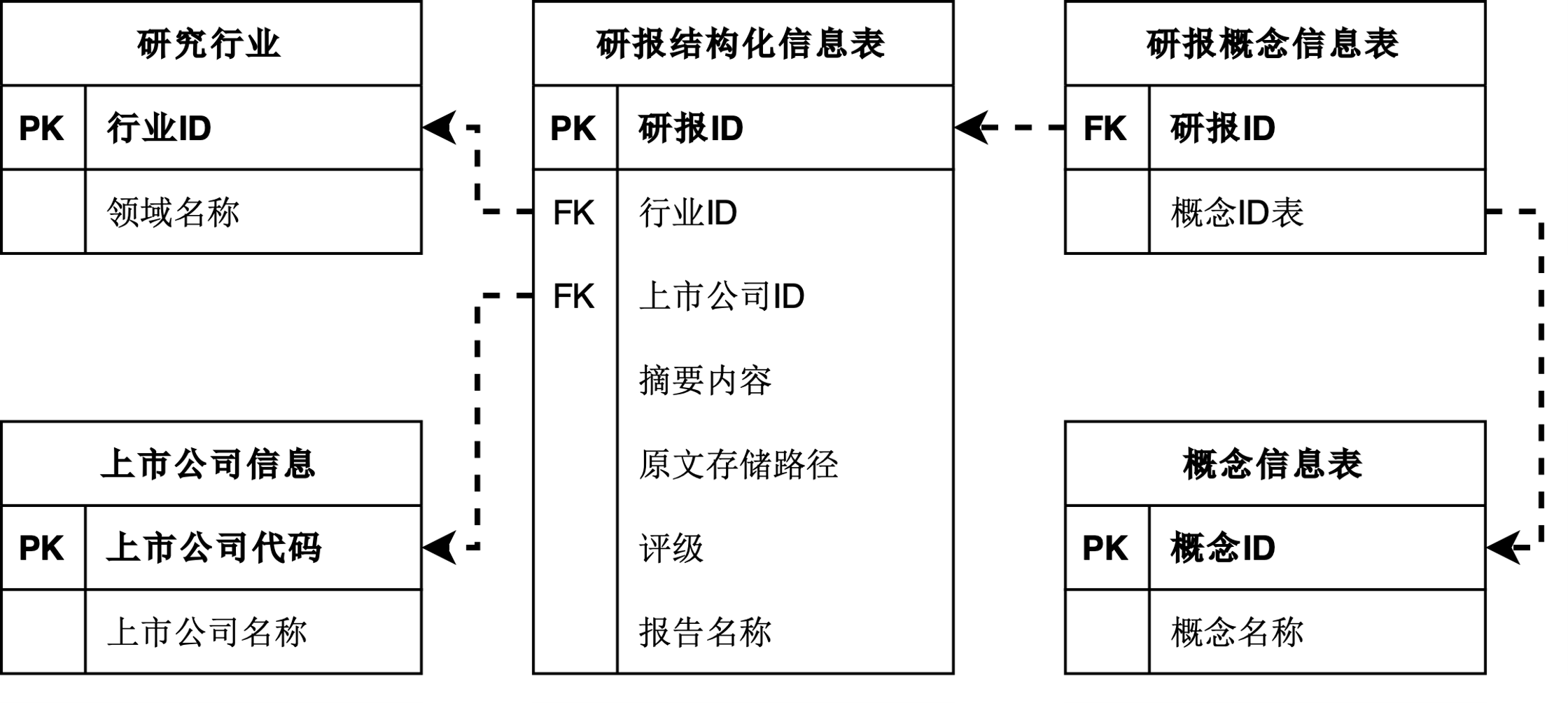
DeltaLake中的表分为三层，Bronze层直接存储原始的采集数据，Silver层则对采集到的数据进行相应的处理和分析，Gold层直接面向数据应用。因此Bronze层的表结构设计会更多考虑数据采集源结构的情况，Gold层的表结构设计会更多考虑应用层使用数据的需求，Silver层则作为Bronze和Gold层之间的过渡进行必须的数据处理和分析。

Bronze表包括两张表，一是行业研报原数据表，二是公司研报原数据表。这两张表在Delta Lake中的表结构设计如下：



这两张表的设计基本按照源数据的原始信息来进行存储。

Silver层的表的内容有两个来源，一个是Bronze层的原数据表，另一个是来自研报PDF原文。



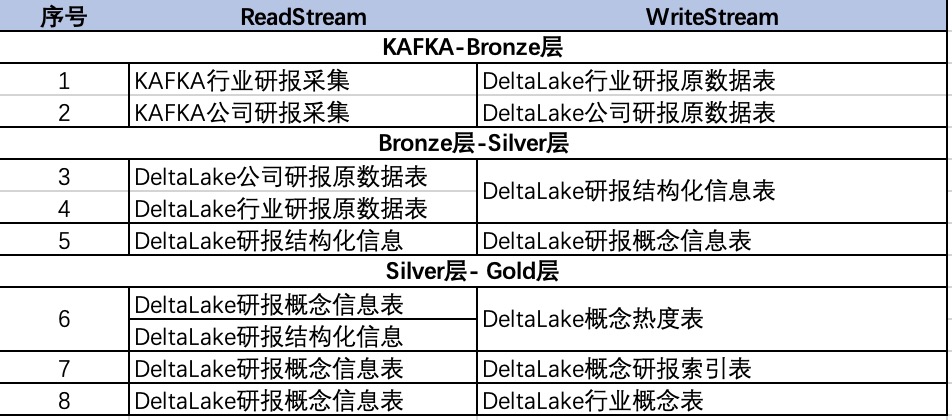
Gold层的表直接面向应用场景，给应用层中的模块提供数据。



**使用Structured Streaming实现持续计算**

Structured Streaming是Spark中实现连续流处理的机制，能够实现数据流的自动流转。外部采集的数据从KAFKA中进入DeltaLake中的Bronze层，以及DeltaLake中各层次表之间的数据流转都是用Structured Streaming来进行处理。

主要需要实现8个Structured Streaming处理单元，列表如下：



从ReadStream到WriteStream转换所需要的计算操作会在计算层中的研报结构化模块和研报热点挖掘模块中进行详细介绍。

### 计算层设计

计算层的作用有两个，一是给数据存储层间的数据转换提供计算支撑，二是为应用层的数据查询提供计算支撑，是将系统中计算功能比较重要的部分模块化后形成的中间层。

#### 研报结构化模块设计

研报结构化模块的主要任务是存储在Bronze层中的原始研报数据转换成可以进行分析处理的数据存入到Silver层中国呢的研报结构化信息表中。

获取到的数据由两部分组成，一是研报的原始文件，以对象存储的形式存储在底层的数据湖中，二是研报相关的信息，存储在Bronze层中的数据表里。

研报结构化的存储包括了三张表，研报结构化信息表、研究行业表、上市公司信息表。研报结构化模块的计算目标是保证存储在这三张表中的内容质量。

三张表中的内容除了摘要内容外都可以从原始信息表中直接获取到，对这部分字段需要检查内容的完整性。对缺失或无法匹配的内容进行修正。

研报结构化信息表中的摘要内容来自于研报原文。由于研究报告的结构一般比较规则，研报首页一般会包含整个研报内容的摘要内容，因此需要一个算法功能来实现从研究报告的首页截取研究报告的摘要内容。

同时，该摘要内容在下一个数据流处理过程中还会被送入研报热点挖掘模块进行处理，从该段摘要内容中分析出研报所包含的概念，因此该算法功能处理后的文本应能够满足热点挖掘模块中命名实体识别功能的要求。

**研报摘要获取算法**

研报文件以PDF格式存储在对象存储之中，通过PDFminer工具对PDF文件进行读取。PDFmine会将PDF首页内容分析为多个TextBox。

研报内容的布局通过清晰的空格进行区分，一般分为左右两侧。左侧为核心内容，包括标题核心观点和摘要的内容。右侧较窄的部分为分析师信息和标的相关信息。

（介绍下PDFminer算法的过程）

PDFminer首先将字符根据参数归类到行，字符间距离小于一定参数的结果会被归类到一行。然后通过行间距离参数来将邻近的行归类为统一个文本块。

对聚合完的文本块可以获取到文本块的高度和宽度等参数，通过找出面积最大的文本块，就可找出首页摘要的具体内容文本信息。

（来一段伪代码）

#### 研报热点挖掘模块设计

**市场概念命名实体识别功能**

经过研报结构化处理后PDF文件形式的研报摘要处理为文本格式，之后便是通过对研报摘要文本的挖掘来识别出其中涉及到的概念。这一问题是标准的命名实体识别任（Named Entity Recognition，简称NER）。但是研报中的市场概念不是通常NER算法所涉及到的人名、地名、机构名等，因此需要针对性的进行语料库的标注。

NER问题的实现算法包括基于隐马尔可夫模型序列标注的命名实体识别，基于感知机序列标注的命名实体识别，以及基于条件随机场序列标注的命名实体识别。同时针对性的预料库规模也对算法的效果有影响。

（这里介绍下过程，分词、词性标注后进行命名实体识别）

对研报内容进行NER识别，首先对转换过来的研报摘要内容进行分词和词性标准，然后将分词后的词语序列特征和词性特征作为NER问题的输入，处理的过程如下：

1. 原文输入：“受益8英寸晶圆紧张，12英寸盈利有望改善”
2. 分词与词性标注处理：“受益/v 8/n 英寸/n 晶圆/n 紧张/v, 12/n 英寸/n 盈利/n 有望/a 改善/v”
3. NER识别结果：“受益/v [8/n 英寸/n 晶圆/n]/np 紧张/v, [12/n 英寸/n]/np 盈利/n 有望/a 改善/v”

分词和词性标注问题是标准的处理过程，不涉及研报特定领域的标注需求，因此这两部分可以使用成熟的第三方库进行解决。

（这里稍微讲一下分词和词性表注的现状吧）

**特征模版**

|  |  |  |
| --- | --- | --- |
| 转移特征 | 词语特征 | 词性特征 |
|  |  |  |
|  |  |
|  |  |
|  |  |
|  |  |

表示当前提取特征的位置，y是结果标签，用{B,M,E,S}的方式来表示，B表示单词为命名实体的开始单词，M表示单词为命名实体的中间单词，E表示单词为命名实体的结尾单词，S表示单个单词为命名实体。是单词本身，表示词性。

**语料标注**

（说明下语料标注的方法和规模）

**三种算法**

**基于隐马尔可夫模型序列标注的命名实体识别**

**基于感知机序列标注的命名实体识别**

**基于条件随机场序列标注的命名实体识别**

**评测方法**

按照惯例引入三个评测指标作为命名四题识别模块的准确率评判标准：

**热度计算功能**

识别出概念后，需要对概念的热度进行计算，采用的算法如下：

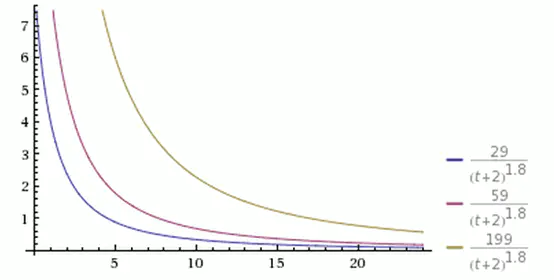
表示热度值；

表示概念的得票数，不同研究机构在不同用户角度是具备不同重要程度的，因此发布机构提出相应概念所产生的票数应该有所不同，对信任的机构应该赋予更多的票数，系统提供相应的管理模块来配置机构的权重；

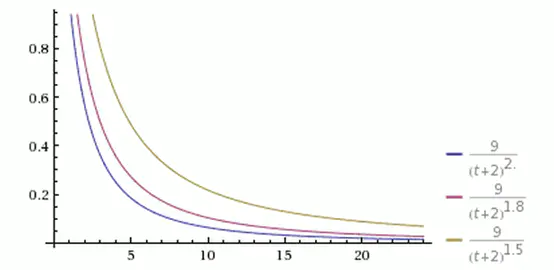
表示距离概念所属研报发出的时间，单位为天，加2是为了防止新发研报导致分母过小；

G表示“重力因子”（gravity power），即将概念热度向下拉的力。

其他条件不变，得票越多，概念热度值越高



重力因子G决定了热度值随时间下降的速度



#### 存储层查询模块设计

存储层查询模块的功能是给应用层提供数据支撑，将面向湖仓一体存储的查询优化集中到该模块中。数据的来源都位于数据存储模块的GOLD层，即概念热度表、概念研报索引表、行业概念表。

上层应用需要查询数据的功能包括查看订阅行业的热点概念、查看热点概念趋势、查看概念相关研报。存储层查询模块通过对外提供相应的数据查询API来给应用层提供数据。设计的查询API列表如下：

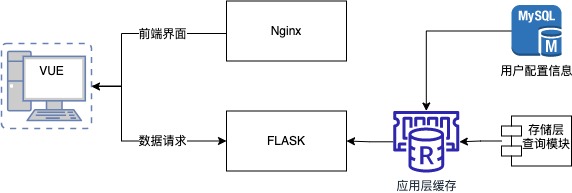
|  |  |  |
| --- | --- | --- |
| **方法名** | **参数** | **描述** |
| getIndustries | 用户ID | 用户订阅行业ID列表 |
| getHot | 行业ID列表、限制个数 | 返回指定行业范围的热点概念ID与热度值 |
| getLine | 概念ID、时间长度 | 返回指定时间范围内的概念热度变化趋势 |
| getReportList | 概念ID、研报数量限制 | 返回涉及该概念的研报信息列表 |

为了提高应用的访问速度满足系统相关非功能性需求，需要对这些查询服务提供缓存机制设计。使用Redis这类的内存KV数据库作为缓存，以查询参数作为缓存数据库的Key。

由于研报信息的获取是以日为单位，因此内存缓存的更新周期以日为单位就可满足缓存与湖仓一体层中GOLD层表数据的一致性。

### 应用层设计

应用层整体采用前后端分离的架构，前端使用VUE作为前端框架，后端使用FLASK作为主要框架进行实现。前端操作界面的分发服务使用NGINX进行分发。整体架构如下图：



除了使用存储层查询模块来查询GOLD层中的数据外，应用层本身也具备一个独立的关系型数据库，用来存储用户相关信息，包括用户密码、权限、订阅的行业等只与应用层实现有关的内容。

#### 订阅热点发现模块设计

订阅热点发现模块包括行业订阅功能、热点发现功能。

行业订阅订阅功能给客户提供管理订阅行业的界面，设计的原型如下：

（行业订阅功能的界面图）用户行业订阅数据存储在应用层独立的用户配置信息数据库中。

热点发现功能是以用户订阅行业研报中所涉及的概念热度值为数据基础的可视化模块。界面分为两部分，一是展示所订阅行业的热点词云，词云展示的概念词数量在系统配置中进行设置；二是订阅行业的热点具体排行列表。

通过点击具体的概念词可以进入该概念词的详细情况界面，展示该概念的发展趋势同时可以看到与概念相关的研究报告。

#### 热点趋势跟踪模块设计

热点趋势跟踪模块展示全市场所有概念的排名变化情况。设计的界面原型如下：

（热点趋势跟踪模块的界面图）

热点趋势跟踪顶部显示了当前排名前10的概念在年度范围内的走向趋势和排名变动情况。排名情况下面显示详细的概念相关信息，同样点击相关概念信息可以进入详细情况界面。

#### 研报阅读模块设计

通过点击概念详细信息中的研报列表可以查看研报的详细内容。使用“PDF.js”作为前端浏览PDF的插件，同时禁止文件的下载和打印功能。

#### 趋势资讯报告生成模块设计

趋势资讯报告生成模块以月为频率生成热点概念相关文章，输出为WORD文件，提供相应的下载界面即可。使用python-docx来进行WORD文件的生成。生成资讯报告文件的排版布局如下：

（资讯报告的排版布局）

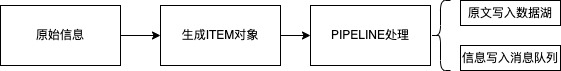
# 系统实现

### 存储层实现

#### 研报采集模块实现

研报采集模块实现了对多个数据源的研报获取。研报采集模块对研究报告主要做两个处理：1.将研究报告原文写入到对象存储中 2.将研究报告相关信息包括存储路径写入到消息队列中触发后续的流式处理。

采集模块使用SCRAPY作为基本框架，针对每个数据源需要实现三部分代码：采集器运行逻辑代码、ITEM处理代码、PIPELINE处理代码。他们之间的处理过程如下图：



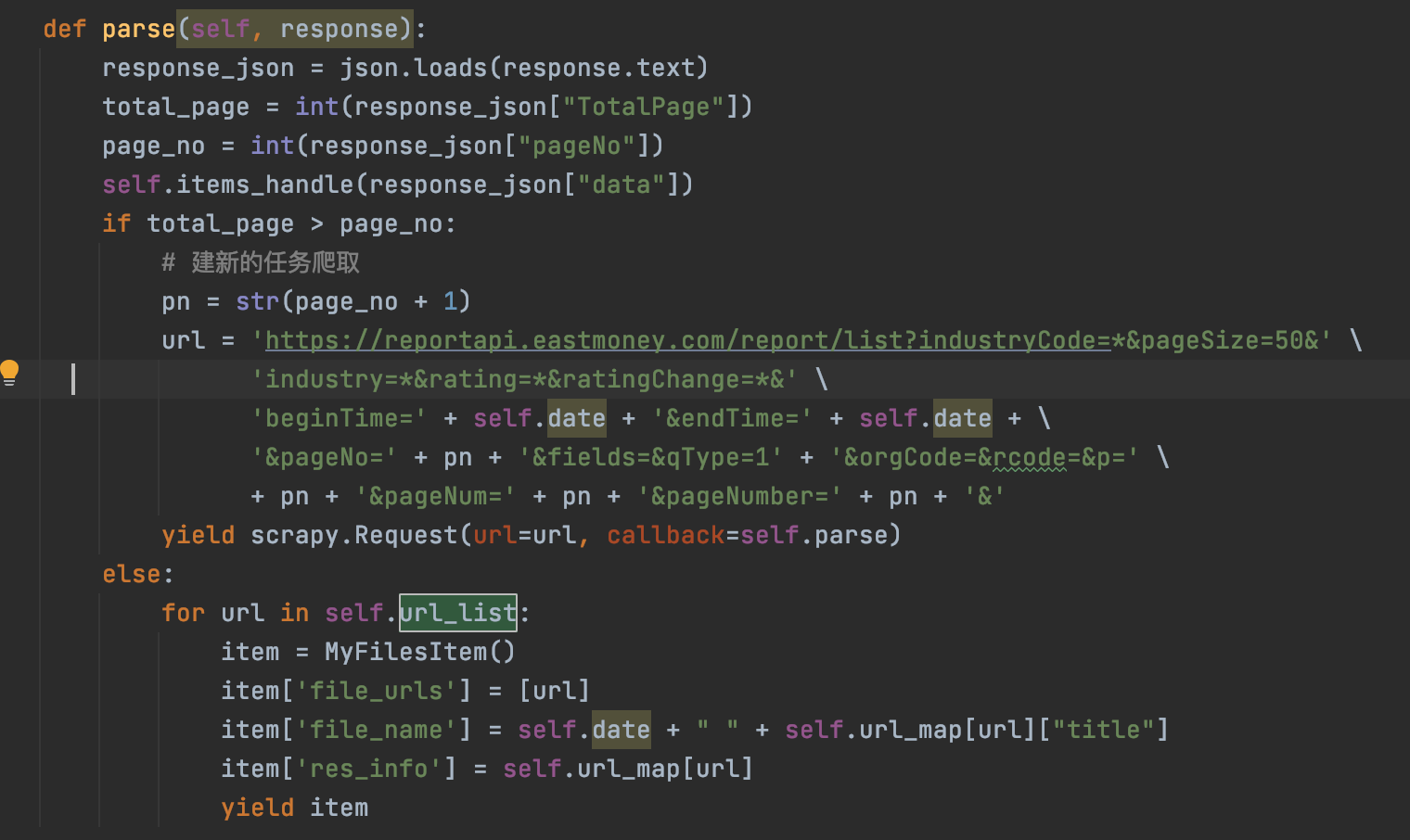
ITEM对象的数据结构定义如下：

class MyFilesItem(scrapy.Item):  
 files = scrapy.Field()  
 file\_urls = scrapy.Field()  
 file\_name = scrapy.Field()  
 file\_paths = scrapy.Field()  
 res\_info = scrapy.Field()

{org\_name":org\_name,"industry\_name":industry\_name,"em\_rating\_name":em\_rating\_name,"last\_em\_rating\_name":last\_em\_rating\_name,"researcher": researcher}

采集器运行逻辑的主要代码示例如下：





函数start\_requests是采集器的启动函数，通过回调parse函数形成循环。Parse函数首先对当前采集到的页面进行处理，将页面原始信息转换成可以用来参数化ITEM对象的信息。判断当前是否是结束页，如果不是就采集下一页信息，如果是就生成所有的ITEM对象并发送给PIPELINE。

PIPELINE收到发送过来的ITEM后，从ITEM中获取URL下载研究报告原文，并把文件写入数据湖中，实现的代码如下：

（下载研究报告并且写入到阿里云OSS数据湖中的代码）

将原文写入到数据湖后将研报数据存储模块BRONCE层需要的数据以JSON格式写入到消息队列中。

（研究报告信息以JSON格式写入到消息队列中的代码）

#### 研报数据存储模块实现

### 计算层实现

#### 研报结构化模块实现

#### 研报热点挖掘模块实现

#### 存储层查询模块实现

### 应用层实现

#### 订阅热点发现模块实现

#### 热点趋势跟踪模块实现

#### 研报阅读模块实现

#### 趋势资讯报告生成模块实现

# 系统测试与验证

## 系统测试概述

## 系统测试环境

## 系统功能测试

# 总结与展望

## 论文工作总结

## 工作展望

# 参考文献

# 致谢