2021年3月1日 星期一

方楠

离职交接文档

一. 交付内容

1. [完成]【ABSC】招投标、资讯邮件服务 spider_bid
2. [完成]【ABSC】中华人民共和国应急管理部 mem_shixin
3. [完成]【ABSC】光曦国际制裁名单 glory
4. [完成]【ABSC】AIC企业舆情 aic_news
5. [完成]【AIC】过往资料及沉淀文件，对接人及与其约定
6. [完成]【AIC】AIC体验平台前端
7. [完成]【AIC】AIC架构设计总纲
8. [完成]【AIC】AIC智能算法部分

9.

二. 【ABSC】招投标、资讯邮件服务

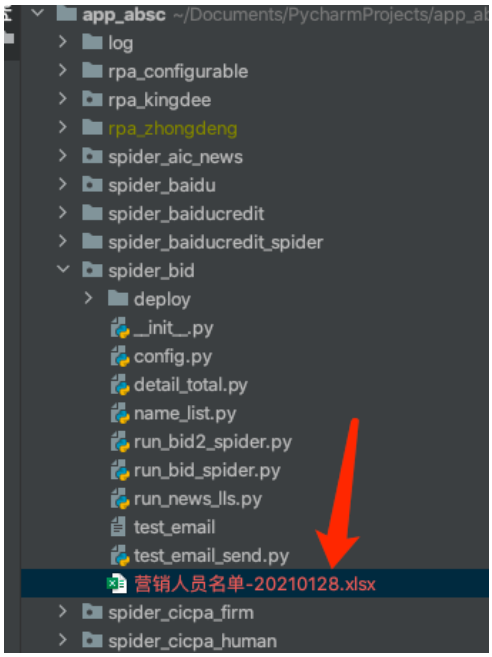
项目信息

项目	内容
名称	招投标、资讯邮件服务
代码包名	spider_bid
代码地址	http://git.hrlyit.com/python_projects/app_absc.git
部署状态	测试环境、正式使用
部署位置	OPS-RPA智能核查平台
业务（需求）负责人	招投标-刘文女，资讯邮件-李如先
企业微信群名	招投标无对应群，资讯【LLS上市舆情采集需求_fang】
接口文档	无

项目概述

招投标邮件是依据刘文女所指定人员（文件：营销人员名单-20210128.xlsx，位置处于代码包根路径，图1）的一个商机挖掘邮件。在每日晚上6点定时发送。发送近期招投标信息给业务人员。

资讯邮件是李如先于1月8日提出监控联易融部分舆情信息，在联易融正式通知静默期开始时启动的需求。每日会从AIC-ES数据库搜索对应数据，组装邮件返回。

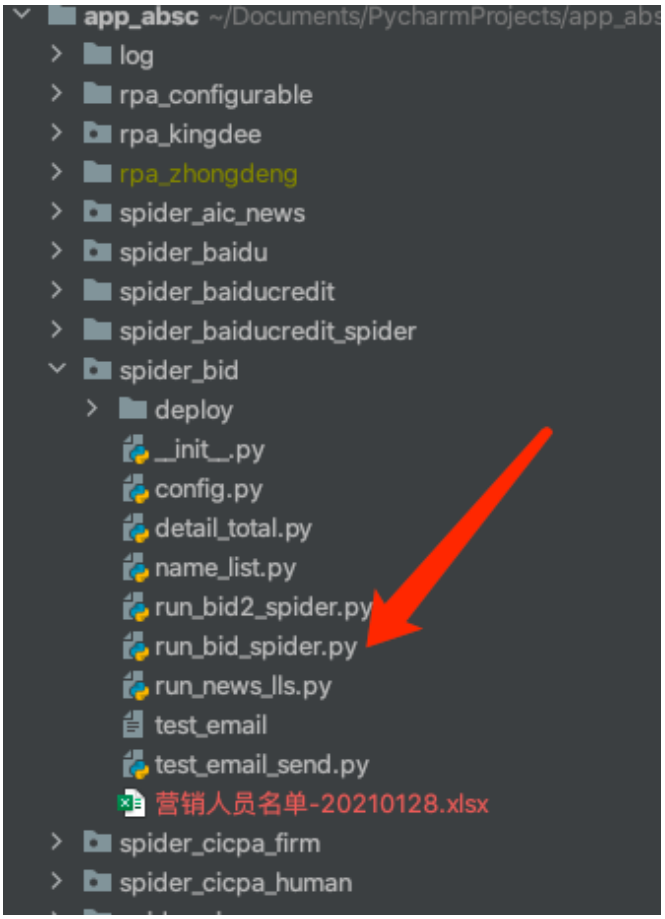


图示：营销人员名单文件位置

项目细节

招投标邮件共计2种，2种会在晚6点的时候同时发。包括V1.0和V2.0版本。

V1.0版本是刘文女提出的四个采集渠道，仅在这四个采集渠道中搜索数据，只搜索标题不含正文。代码逻辑是一个定时器死循环采集四个网站中的当日最新数据。组装成邮件存入MongoDB。定时器可随时修改时间，定时器触发的时候将最新一条存入MongoDB的邮件发出。



图示：V1.0招投标代码位置

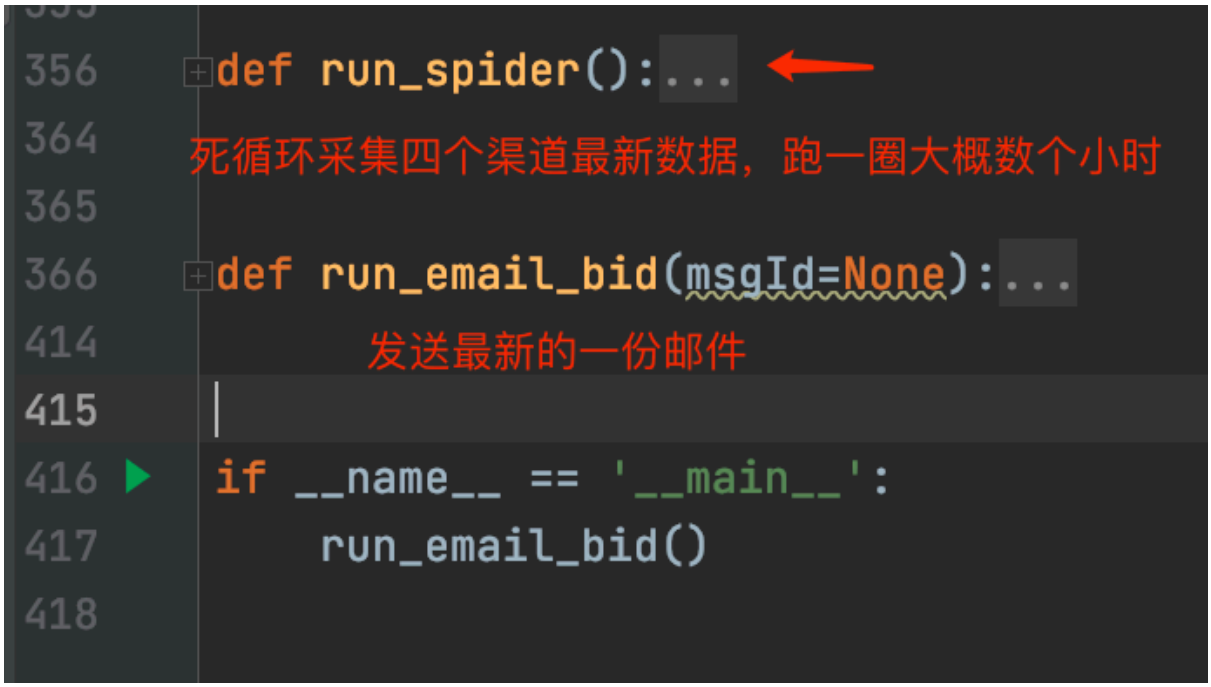
```
# 非代理方案 全国公共资源交易平台  http://deal.ggzy.gov.cn/ 有翻页 接口
def getGgzy(keyword, start_time, end_time):...


# 代理方案 招标与采购网  https://www.zbytb.com/ 有翻页 网页
def getZbytb(keyword, start_time, end_time):...


# 代理方案 全国招标信息网 https://www.bidnews.cn 有翻页 网页
def getBidNews(keyword, start_time, end_time):...


# 代理方案 中国招标投标公共服务平台  http://www.cebpubservice.com 有翻页 接口
def getCebpubservice(keyword, start_time, end_time):...
```

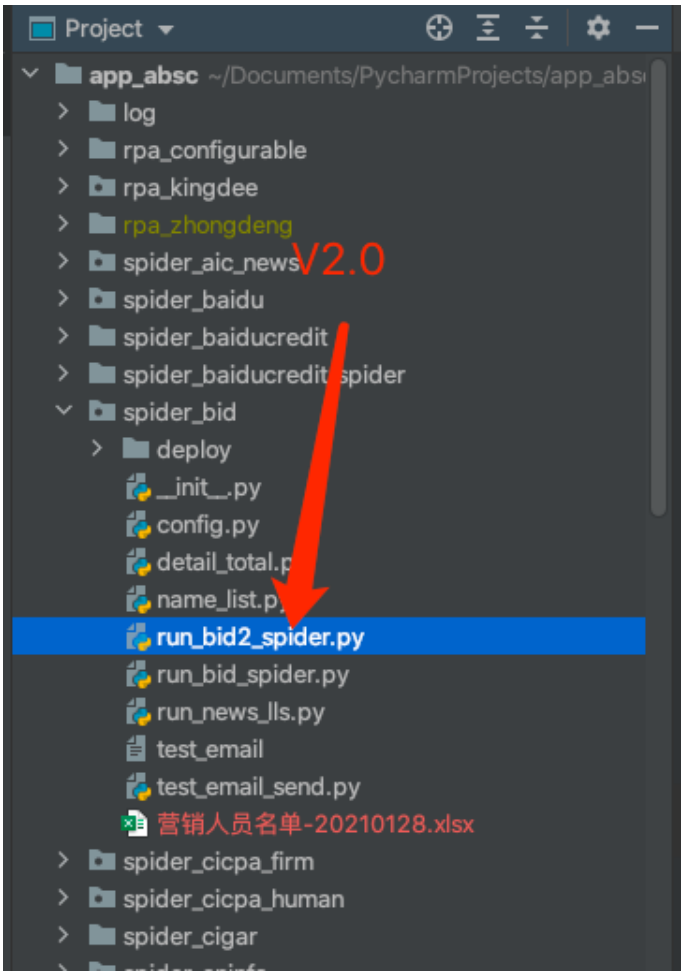
图示：V1.0 采集的指定的四个渠道-采集代码



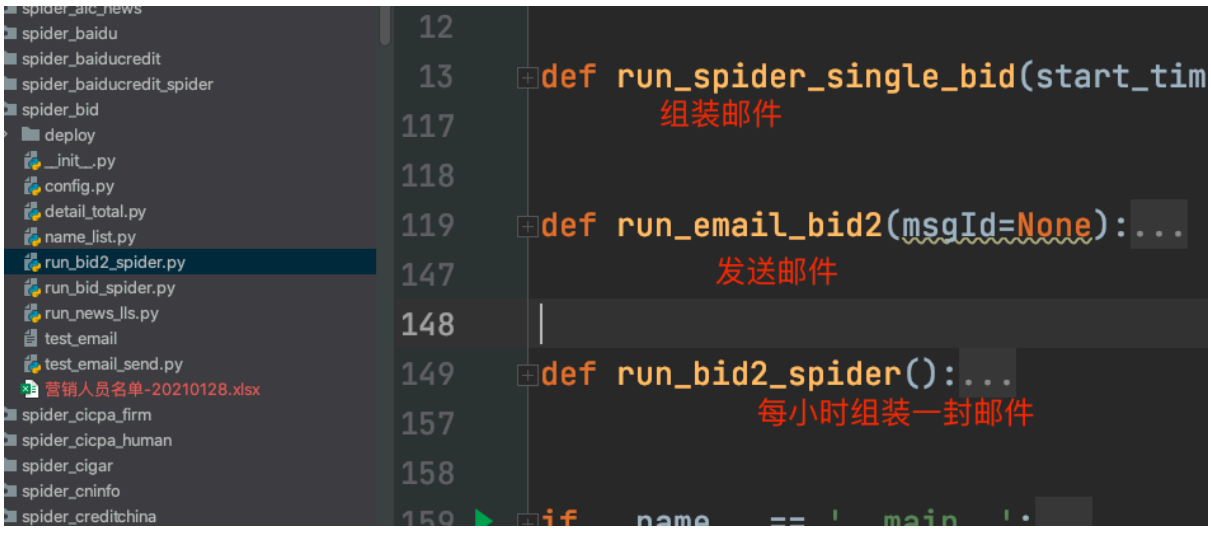
图示：业务执行代码

V2.0版本是通过AIC-elasticsearch数据库，召回AIC项目采集的招投标数据。V2.0相对于V1.0来说，优势是渠道数大大增加、搜索策略可通过ES-SQL随时调整和定制化，返回内容带正文。因为是引用AIC项目的数据，因此不存在采集部分，只有组装邮件和发送部分。采集部分由刘言维护负责，是其2021年1月-3月的主要工作。

虽然没有采集部分，但是架构继承了V1.0的设计。每60分钟会组装一封邮件入mongoDB，无论什么时候发送，就发送最新的一份。



图示：代码位置



图示：业务执行代码

遗留问题或在做需求

业务希望V1.0和V2.0合并至一起，形成只有一个V2.0。因此需要做三个事达到以下目的：

（1）需要将aic招投标的数据采集开发的更加稳定、多增加招标信息的采集侧重。

（2）结合需求去分析，优化ES-SQL的搜索逻辑（现状是只查询标题中是否包含）

（3）AIC添加V1.0中涉及的4个渠道。

三. 【ABSC】中华人民共和国应急管理部

项目信息

项目	内容
名称	中华人民共和国应急管理部
代码包名	spider_mem_shixin
代码地址	http://git.hrlyit.com/python_projects/app_absc.git
部署状态	生产环境、正式使用
部署位置	OPS-RPA智能核查平台
业务（需求）负责人	刘维静-渣打项目
企业微信群名	渣打版本-对接联调
接口文档	https://www.mem.gov.cn/fw/cxfw/xycx/hmdgg/

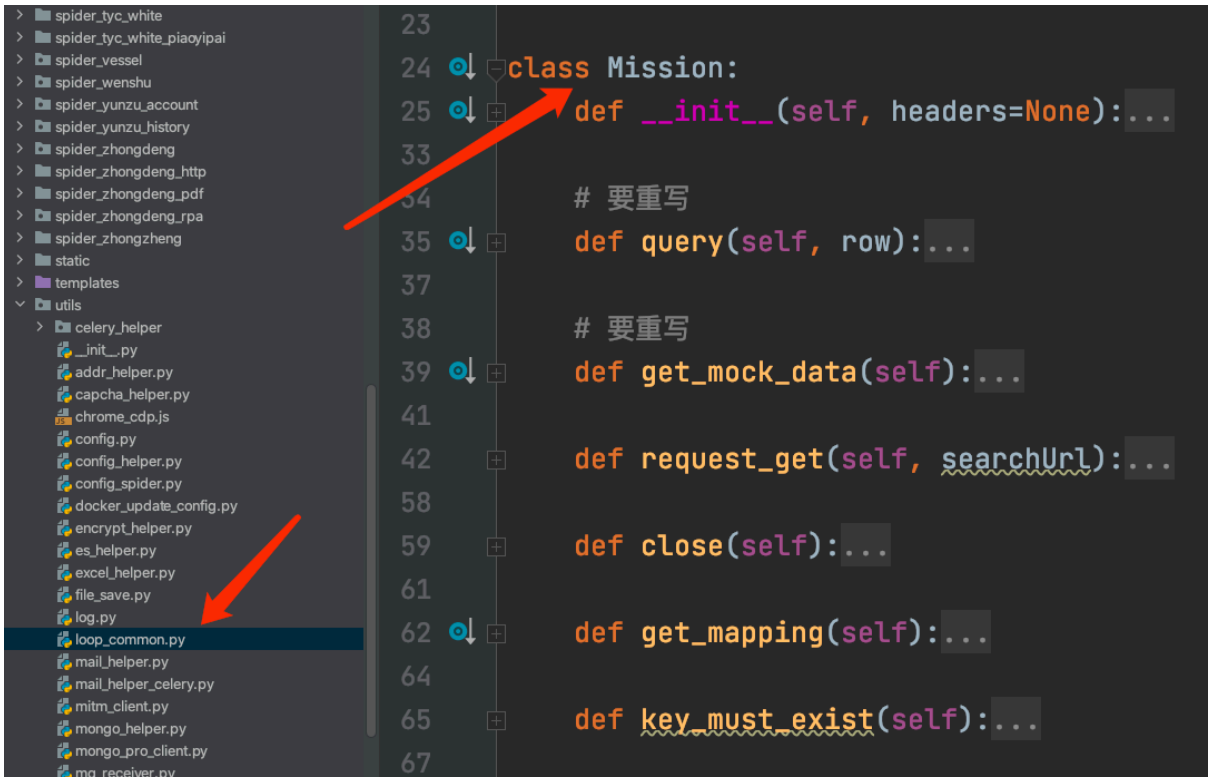
项目概述

渣打风控接口，详情参见接口文档。已在生产环境上线

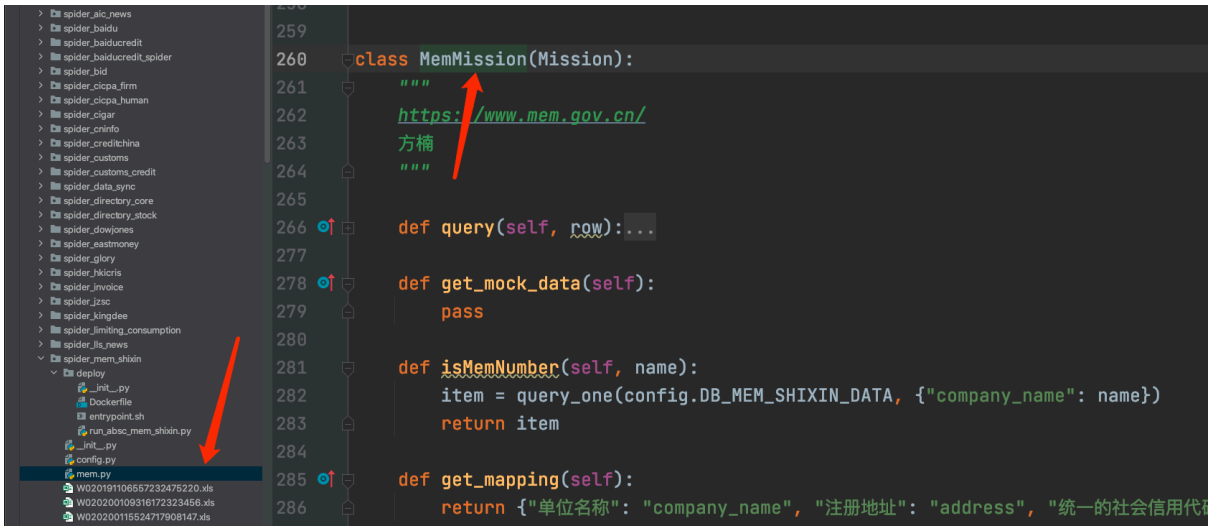
项目细节

ABSC的MISSION封装

对于absc我涉及开发的项目，都采用了一个基于Mission抽象类的封装。封装完成后，使用loop_data_new函数启动，即可自动接入【冼东亮】的MQ-absc主流程。对于该主流程涉及的监控、异常下发、生产测试环境切换、联调等问题。可咨询【冼东亮】、【王金辉】或【刘言】。



图示：被继承的抽象类MISSION



图示：继承它后重写QUERY方法，其他的可写可不写

图示：正式使用，INSTANCE作为继承抽象类的实例化对象传入

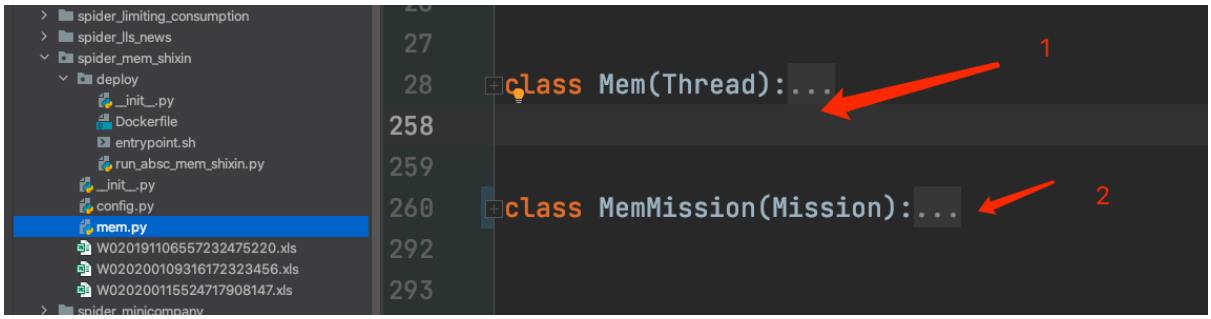
特别说明一下抽象类的get_mapping方法，应对采集回的结果的key经常需要修改的痛点（比如中文等场景）。为了加快字段编写速度，可使用【刘言】开发的Dict英文翻译器。将范例dict自动转化为翻译后的字段，直接将结果粘贴至get_mapping中，加速字段编写速度。

```
def get_mapping(self):
    return {
        "单位名称": "company_name",
        "注册地址": "address",
        "统一的社会信用代码": "company_code",
        "主要负责人": "people"
    }
```

图示：一种GET_MAPPING的使用方式，如果不使用，返回{}即可

本业务为采集查询异步型任务，Mem类为去指定网站下载Excel，解析Excel数据后入库。特别说明，有几个Excel不知什么原因，xlrd无法解析报错，只能手动下载另存为后即可正常解析。好在该网站的excel大约一年更新1-2次。如果出现解析不了的excel手动处理一下即可，大部分都是可以解析的。

Mem类以定时器的方式运行，每天运行采集一次。随后MemMission类继承了Mission抽象类。简单的去mongoDB查询一下返回结果。



图示：类位置

四. 【ABSC】光曦国际制裁名单

项目信息

项目	内容
名称	光曦国际制裁名单
代码包名	glory
代码地址	http://git.hrlyit.com/python_projects/app_absc.git
部署状态	生产环境、正式使用
部署位置	OPS-RPA智能核查平台
业务（需求）负责人	刘维静-渣打项目
企业微信群名	渣打版本-对接联调
接口文档	https://autotest.lianyirong.com.cn/project/394/interface/api/202766

项目概述

渣打风控接口，详情参见接口文档。已在生产环境上线

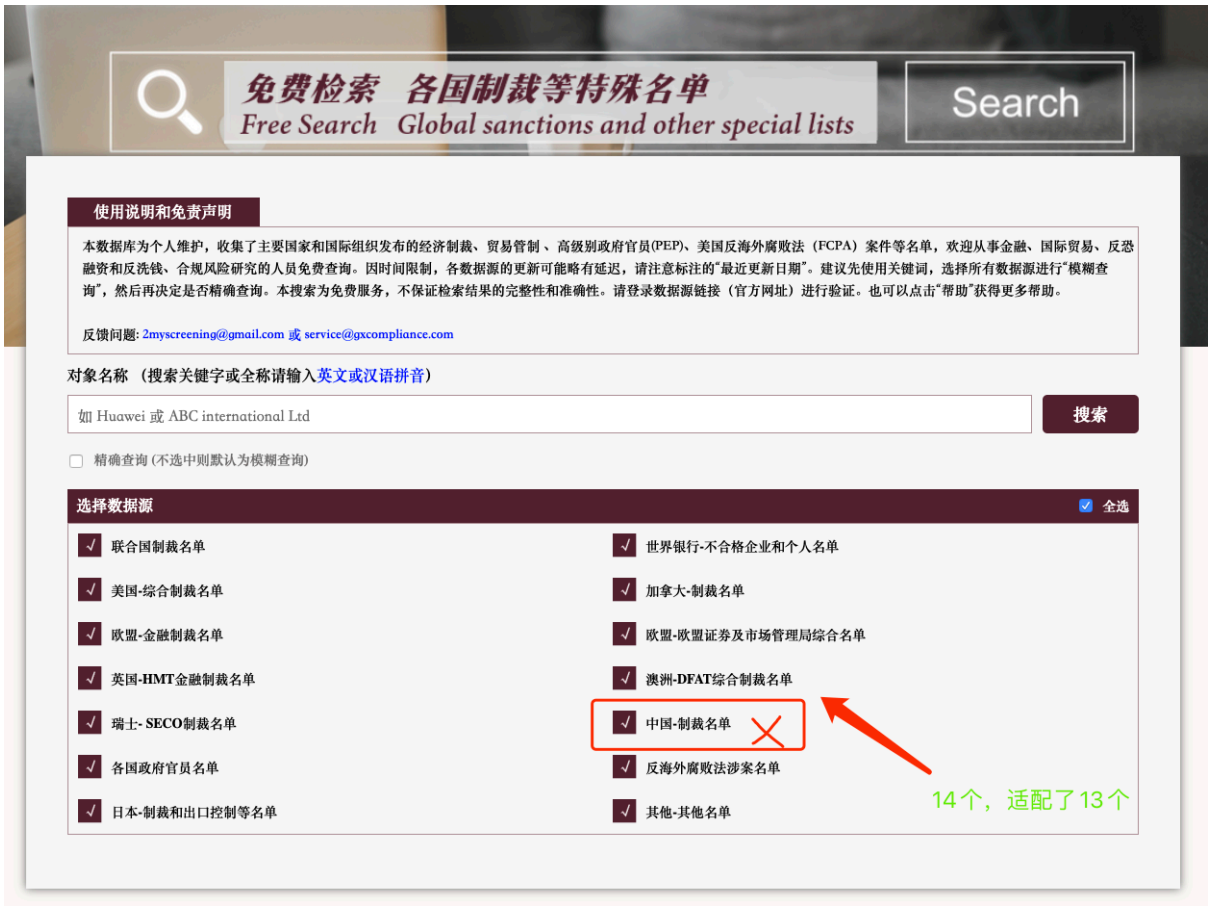
项目细节

若未阅读第九页《ABSC的MISSION封装》，可能会存在疑惑。请先阅读该章节再看实现代码。

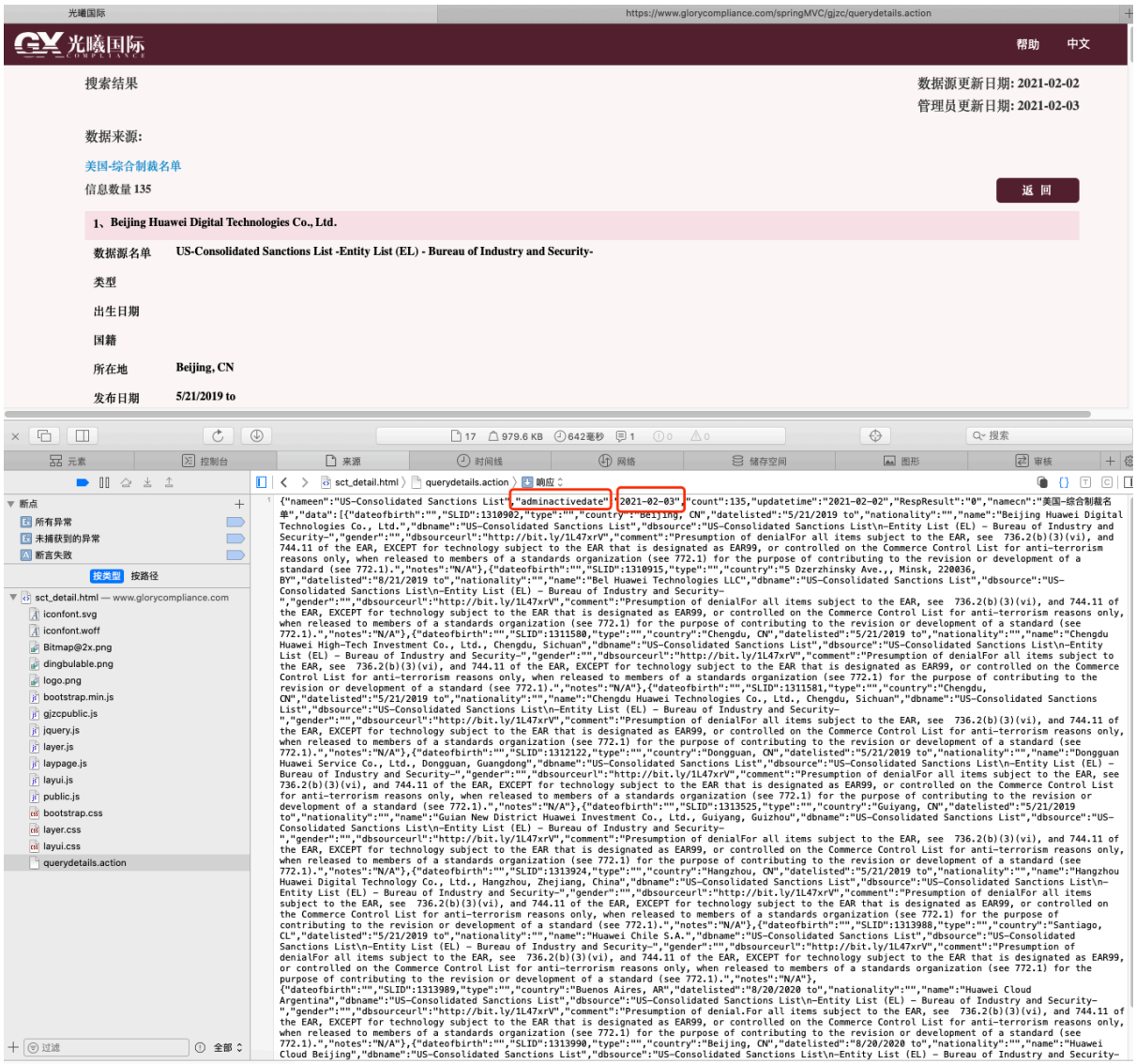
搜索网址:https://www.glorycompliance.com/springMVC/webpage/html/sctsearch/index.html

接口网址:https://www.glorycompliance.com/springMVC/gjzc/querlist.action

该数据现在有14个可能，现适配了13个，未适配《中国-制裁名单》，原因是找不到中国制裁的企业。由于全部为英文字段，现在字段是自然的英文，因此Mission中get_mapping无任何映射。如未来需要增加，在get_mapping中增加映射字段+修改文档即可。



图示：14种返回的数据结构



图示：其接口返回内容本就是JSON结构，且KEY全为英文，无需再做中文->英文映射，只需写到接口文档中即可。

五. 【ABSC】AIC企业舆情

项目信息

项目	内容
名称	aic企业舆情
代码包名	aic_news
代码地址	http://git.hrlyit.com/python_projects/app_absc.git
部署状态	生产环境、正式使用
部署位置	OPS-RPA智能核查平台
业务（需求）负责人	刘维静-渣打项目
企业微信群名	渣打版本-对接联调
接口文档	https://autotest.lianyirong.com.cn/project/394/interface/api/263828

项目概述

业务对接AIC的唯一接口，详情参见接口文档。

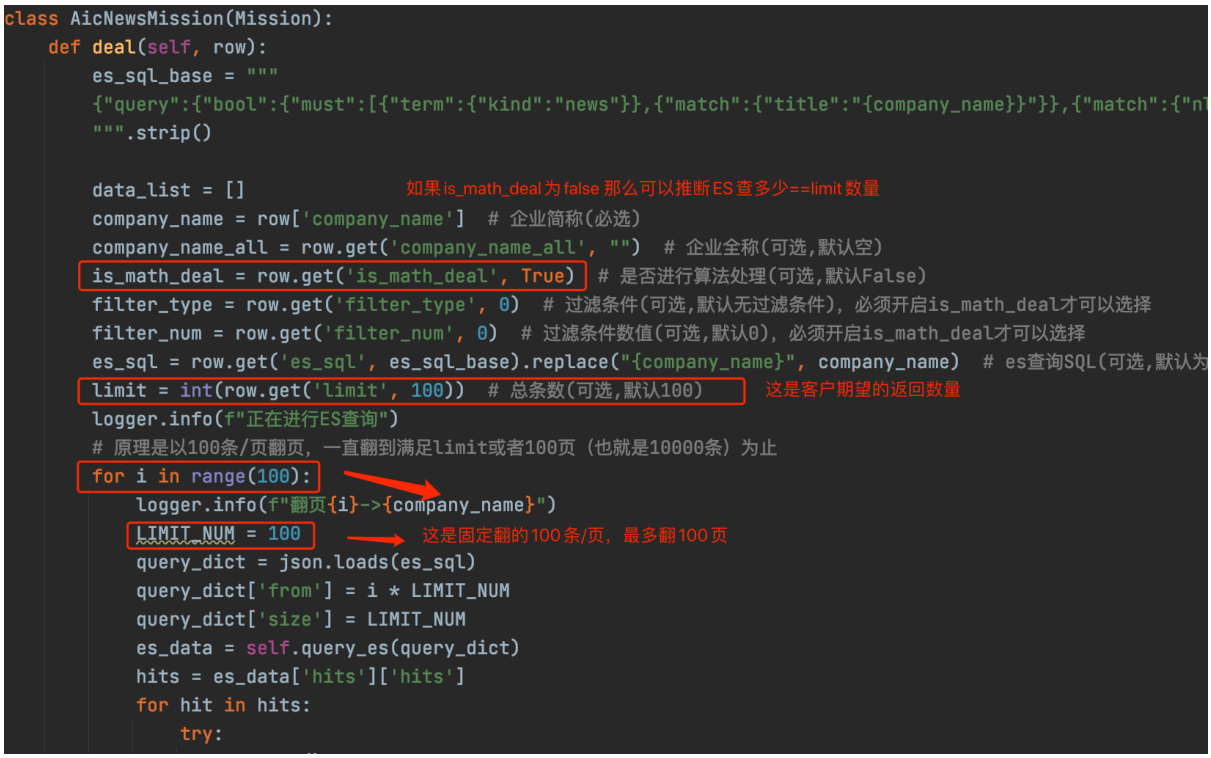
项目细节

该接口不涉及任何爬虫业务，只有【读取ES】和【调算法】接口两个核心功能。

算法接口：http://172.16.86.38:8115/api/ocr/v1/news_relsen_server（维护人王伟）

这里的逻辑稍显复杂。接口会传入一个limit，代表他期望返回的数量。比如说是500。但是不代表直接从ES中查500就可以了，因为还需要返回算法返回与目标公司相关的新闻（不相关的不要）。因此可能需要查700个，其中200个不相关，500个相关。

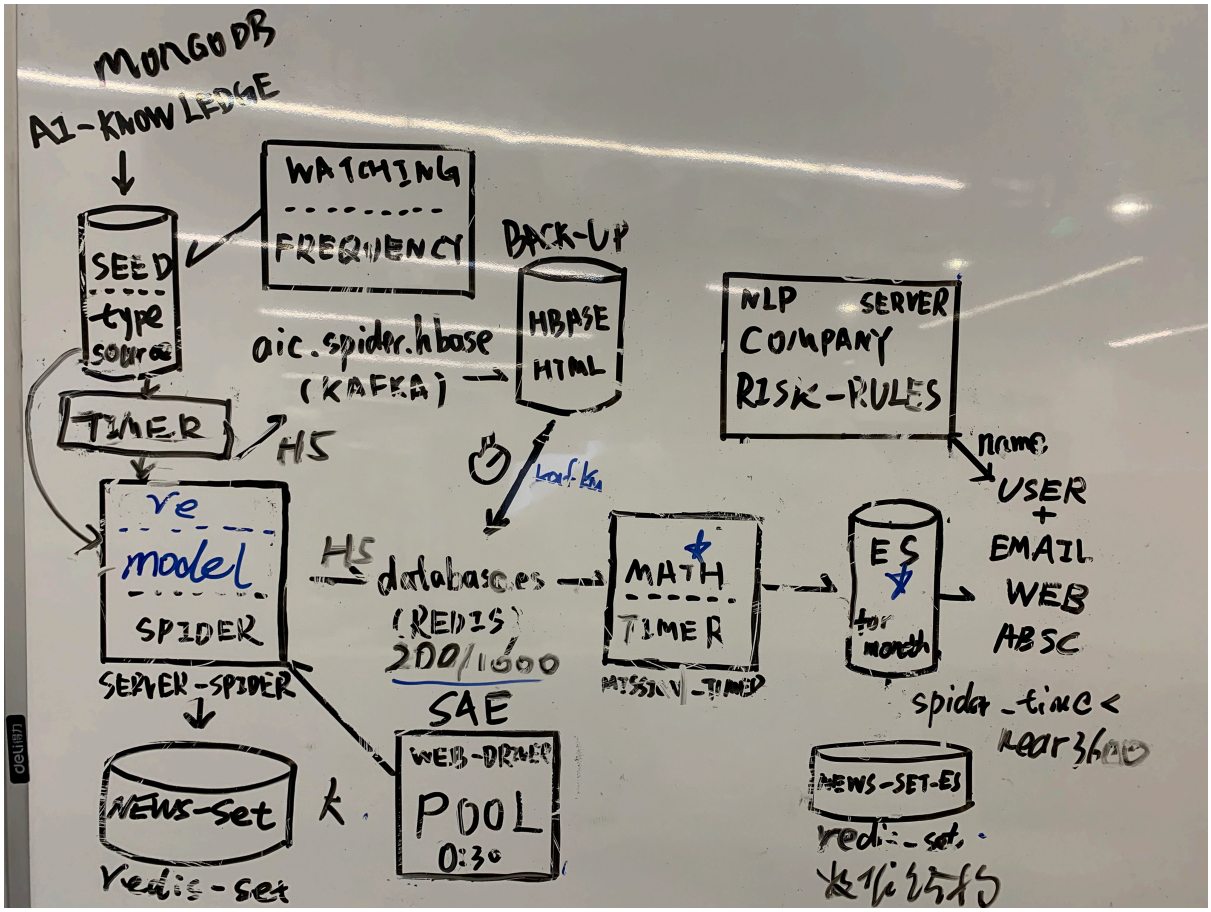
根据上述表述，可以理解ES查询的数量>=客户需求的limit。因此无论客户limit写多少，代码中都以100条/页的条件去查，最多查100页。如果查的过程中返回的总数达到了客户的要求limit或者到达第100页，则返回结果。



图示：由图可见，客户期望返回100条与目标公司相关的新闻，考虑到算法不一定返回相关，查询应>=100条。

六. 【AIC】架构设计总纲

架构图



图示：白板上的架构图

模块列表

序号	介质	名称	功能	负责人
1	mongoDB	seed表	渠道列表、种子下发、监督模型依赖数据。所有的任务下发都是从Seed表开始获取渠道URL的。	刘言
2	Python-ops	Watching-Frequency监测系统	监测数据在体验平台上显示。监测包括每个渠道的每次具体的采集时间、当次采集入库数，以及如果没有入库的错误原因。	刘言
3	Python-ops	Spider	数据采集核心模块，由内置的三级抽象【spider】【model】【math】共同组成。	刘言
4	Redis-db26	news_set去重表	爬虫模块公用URL去重redis-set。由爬虫代码维护并使用	刘言

5	Redis-db26	database_es队列	mission_timer指定消费队列，用来存储爬虫采集下来的包括html5的具体数据内容。特别的，该队列存在200/1000设计。重跑数据最多占用队列200个任务，最新采集数据最多占用1000个任务。如超过会自动舍弃旧的。	方楠
6	driver_pool	SAE	spider调用的webdriver采集器，由智恒负责维护。	庄智恒
7	Python-ops	math_timer	又常称作mission_timer。消费爬虫采集数据进行统一解析处理。是OPS上开销非常大的容器实例。	方楠
8	hbase	Hbase_kafka	由大数据组负责存储全量未经处理的HTML数据，爬虫采集成功后先向该Kafka发送消息。避免数据由于其他原因未能跑通存留备份	谢智豪
9	python_local	NLP	由算法组负责进行基于业务的新闻相关度识别接口、规则命中接口	王伟
10	ES	es	存有全量新闻结构数据。	方楠
11	Redis-db26	news_set_es	Es的去重库，由于es是按月分表，表与表之间无法去重，因此整体引入去重set进行去重	方楠

七. 【AIC】过往资料及沉淀文件，对接人及与其约定

附件列表

编号	文件名	功用
1	2020年年度总结	个人绩效及产出
2	词性规则库V0.1	过往规则参考版本，现已有最新版本，即编号3
3	情报中心相关舆情规则20210126	规则最新版本，是编号2的升级版
4	资讯AI智能采集理想计划3期	3期内容中留存了1期、2期的所有内容
5	AIC智能采集会议纲要	2期完结汇报的思维导图
7	AIC_NEWS_SEED备份	mongoDB中的一个备份表

八. 【AIC】AIC体验平台前端

项目概述

AIC体验平台用以做领导展示、功能体验、异常报警、数据统计和测试功用。分为全景数据、渠道监控、ES招投标、ES新闻舆情四个模块。

图示：四个功能模块

项目打包与启动

首先初始化加载 npm install(以下指令都在根目录运行)

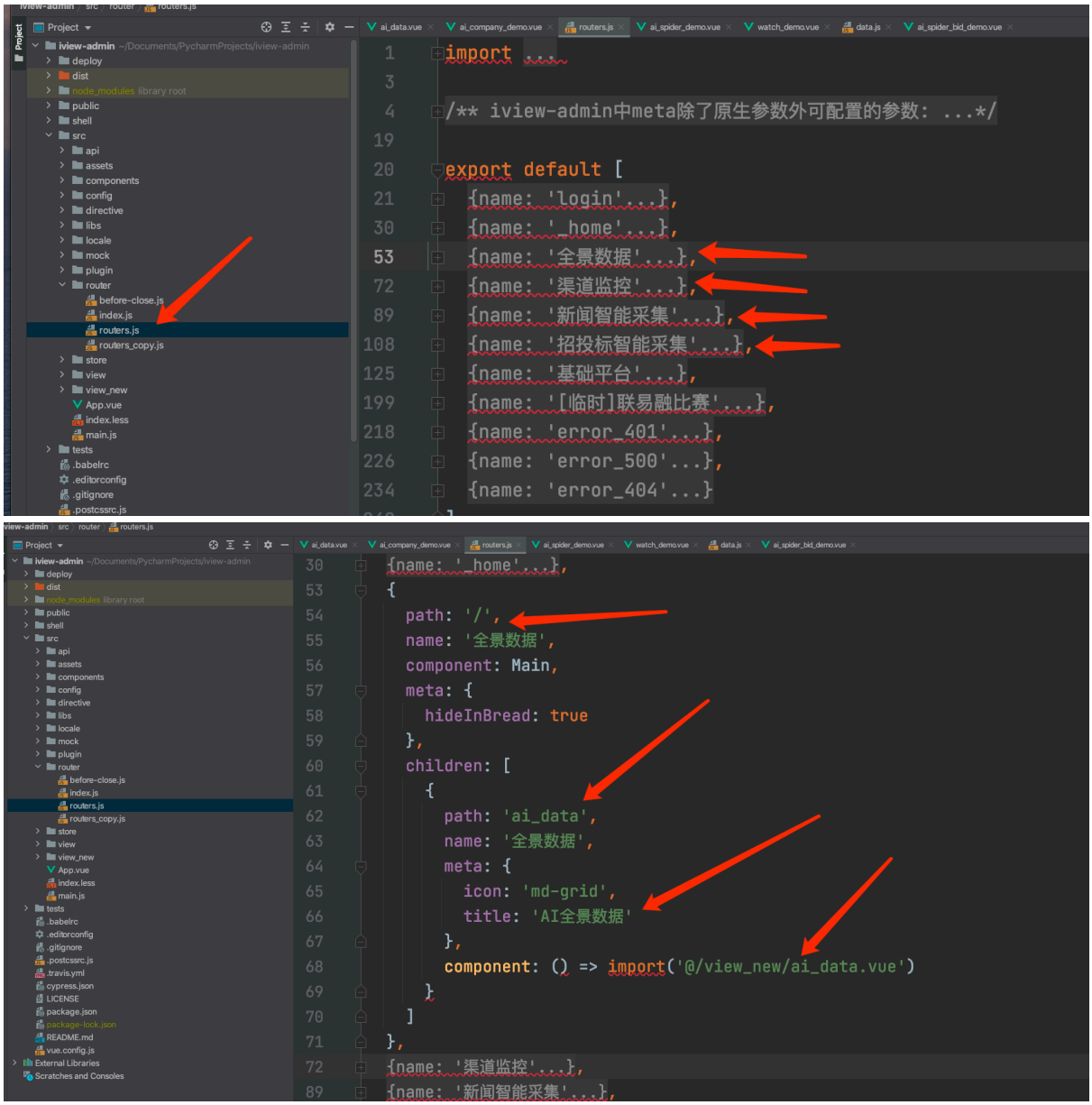
本地测试指令 npm run dev

本地打包指令 npm run build

打包完成后，连带打包生成的dist上传git，ops即基于nginx容器进行了静态资源部署上线。

项目路由位置

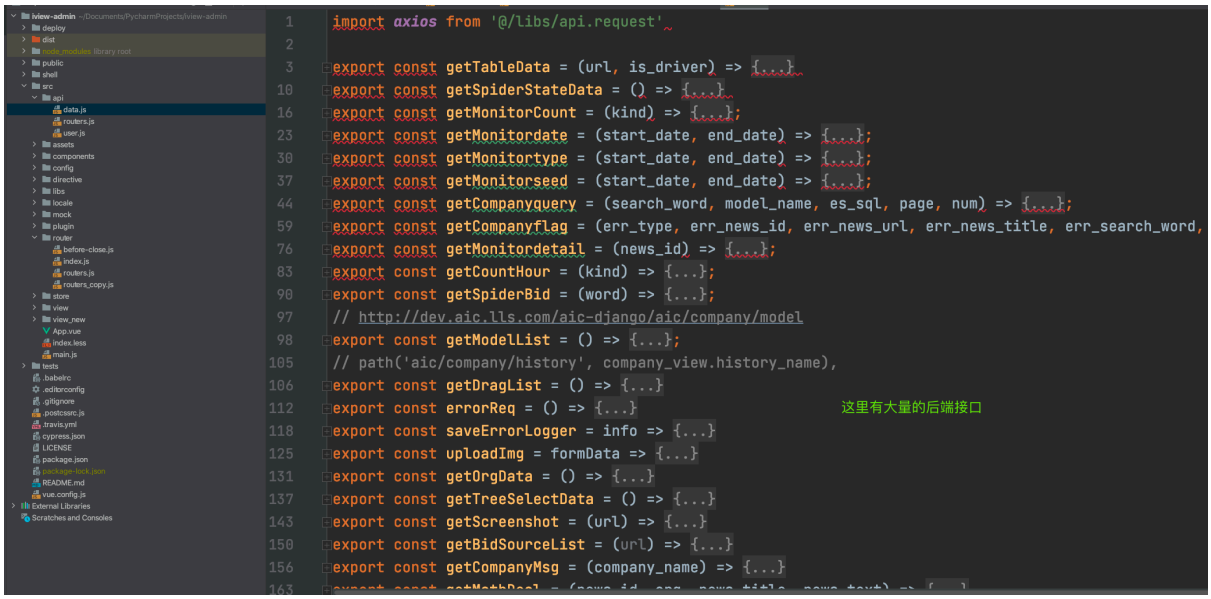
src/router/router.js 配置路由，特别注意，由于本业务是由nginx部署的，为了防止路由设置与nginx有冲突，所以使用hash模式。



图示：路由位置及一个模块的举例

接口位置

src/api/data.js 使用axios组建进行HTTP请求。如果想修改相对路径前面的请求头，可修改src/config/index.js中的baseUrl变量。



图示：HTTP请求接口列表

全景数据

提示，如果有一些功能需求，可以直接拉https://github.com/iview/iview-admin.git看里面内置功能。

位置：src/view_new/ai_data.vue

这个比较简单，结合着视图看就行了。大量的echart折线图饼图。特别提醒，注意created、watch和mounted的顺序，以及使用api之前要import引用

渠道监控

位置：src/view_new/watch_demo.vue

功能：这个属于aic watching_frequency 模块，用研发来渠道查错。

ES-招投标

位置：src/view_new/ai_bid_demo.vue

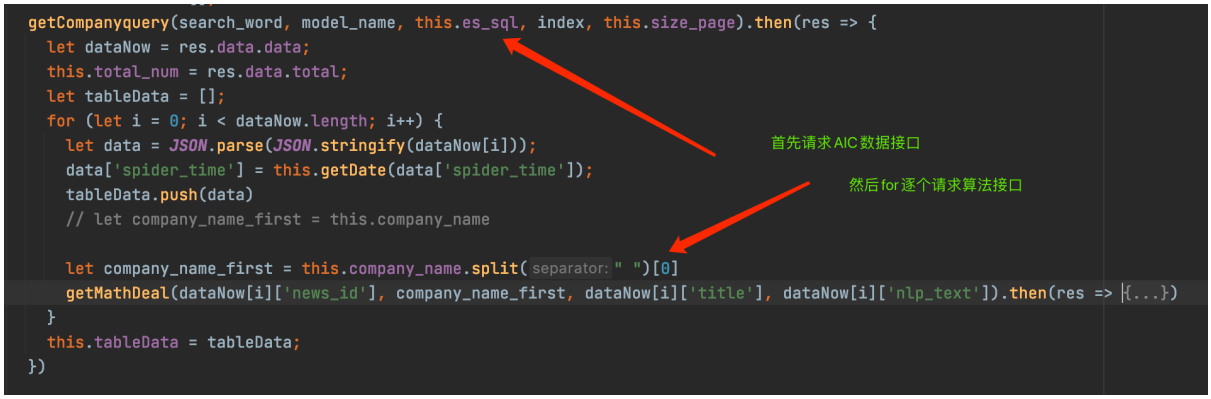
功能：体验招投标数据采集功能，可以查看详情，关联天眼查，打错误标签从而进行测试。

ES-新闻舆情

位置:src/view_new/ai_company_demo.vue

功能：体验新闻数据采集功能，可以查看详情，打错误标签从而进行测试。

尤其要注意双重异步的这个结构。首先是一次性批量获取该页所有新闻，调的是aic的接口，一次性渲染所有数据，但是算法的【情感分】和【相关性】两列为空。随后再异步、逐个调用算法接口，根据调用算法接口的news_id渲染指定位置的数据。



九. 【AIC】智能算法部分

项目信息

项目	内容
名称	Aic智能算法部分
代码包名	base.math_utils
代码地址	http://git.hrlyit.com/fangnan/aic_collector.git
部署状态	测试环境、正式使用
部署位置	OPS-AIC智能采集
业务（需求）负责人	刘言-数据采集依赖
企业微信群名	资讯AIC智能采集项目

项目概述

通过通配模板、抽取算法等方式，使得爬虫采集代码无需写任何re\css\xpath即可完成采集流程，一次采集多渠道适配。提升采集效率。

算法之间可能存在复杂的引用和嵌套。python的引包策略保证算法实现类不存在循环引用。

项目细节

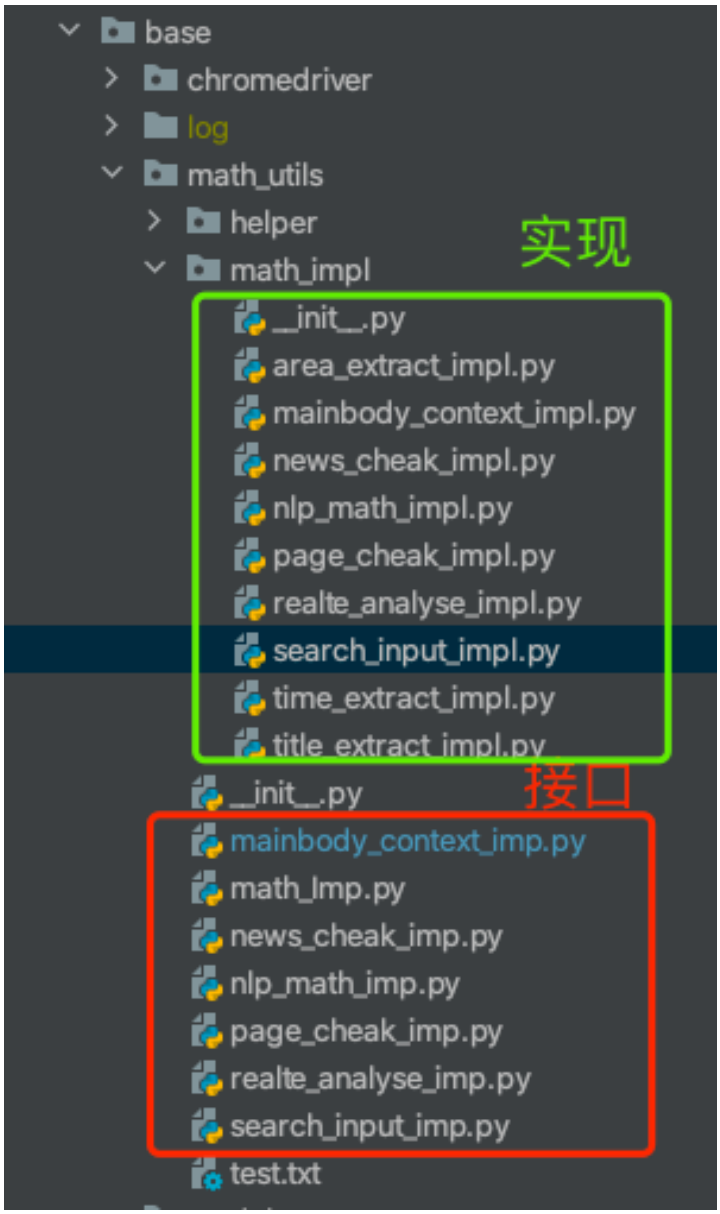
算法接口类

现在共存在7个算法接口，用以解决不同的问题。一个接口可能有多种实现，实现接口的方式包括但不限于开发新算法、改造旧算法、拼接或组装原有算法等。每一个算法接口都有对其输入值、输出值的定义。输入值在函数入参中定义，输出值在接口上方的class_bean中定义。

另外特别说明，一个算法不用实现bean中所有的字段，可以只实现部分字段。

序号	接口名	用途	说明
1	MathControlImpl	这个是所有接口的接口抽象，以下6个所有接口都要继承该接口	通用的方法写在一起，节约时间
2	MainBodyContextImp	抽取一篇文章中的正文部分	存在两种bean，新闻和招投标。招投标继承新闻的bean，有新闻的全部字段。
3	NewsCheakImp	寻找一个中间页中所有的详情页链接	针对a标签中非链接的特殊采集情况，接口Bean返回None，让model层自行写逻辑处理

4	NLPMathImp	文章特征提取	重要的是地点和公司名，别的从业务角度来说不太重要
5	PageCheakImp	翻页算法	返回所有翻页url。特别注意的是，有可能返回超过个数的页（比如，一共只有3页，但是可能返回十个url，有七个都是不好使的，一个宁多勿少的逻辑）。继承的实现首先保证其“不漏”为首要条件，然后尽可能的准确。
6	RealteAnalyseImp	公司相关度算法	输入一个公司名，查询该文章是否与输入的公司名有关。现在接口为废弃状态，用NLP提供的查询HTTP接口代替。尚未继承实现
7	SearchInputImp	搜索框位置定位	寻找网页中搜索框位置
9	TimeMainbodyContext	时间抽取算法	主要基于正则完成
10	TitleMainbodyContext	标题抽取算法	主要基于标签完成



图示：接口和实现类的具体位置

算法继承实现

继承实现一般分为【有监督】和【无监督】的两类模型。有监督是指该算法需要从mongoDB或其他表中，根据渠道有定义好的特征（如URL正则等），拿取该【监督信息】进行算法分析，换言之，如果该渠道不存在监督信息，则无法使用有监督算法。

无监督算法恰恰相反，无监督算法不需要监督信息。基于采集数据就可以进行分析。往往使用有监督的原因有下面几种可能：

（1）该算法实现起来过于艰难，需要【监督信息】的帮助，而这个【监督信息】获取的成本相对较低。

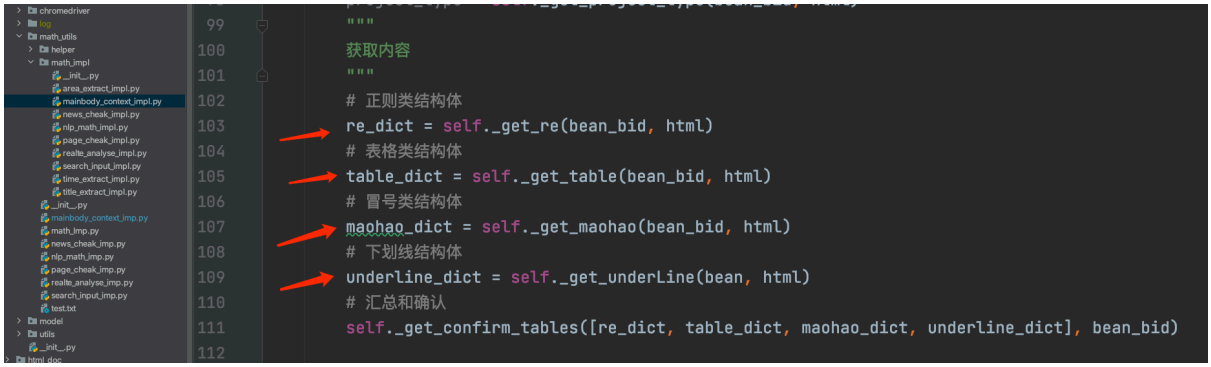
（2）该算法性能消耗太高导致运行时间太长。把一些中间量提前缓存成【监督信息】有助于提升性能。
（3）历史遗留问题，已经有成熟的【监督信息】，不用白不用。先跑起来再说。

接口的实现要严格按照接口类的定义，但并非我们要实现接口类设计Bean的所有字段。因此我们可以根据业务需求添加Bean中的字段使我们的算法的应用面更加广泛。一般来讲我们都会定义一个Common开头继承类作为【推荐】使用类。一般爬虫model不出意外的话绝大多数情况都会使用Common实现方法。接下来会对所有涉及到的接口实现类进行说明：

mainbody_context_impl（重点）

CommonMainbodyContext用于新闻采集正文提取，主要使用的n3k作为主要算法。jieba线上未调通（jieba调用会出错），所以per\loc\org都为空，本质上是没有打好的（但jieba报错异常不会影响主流程）

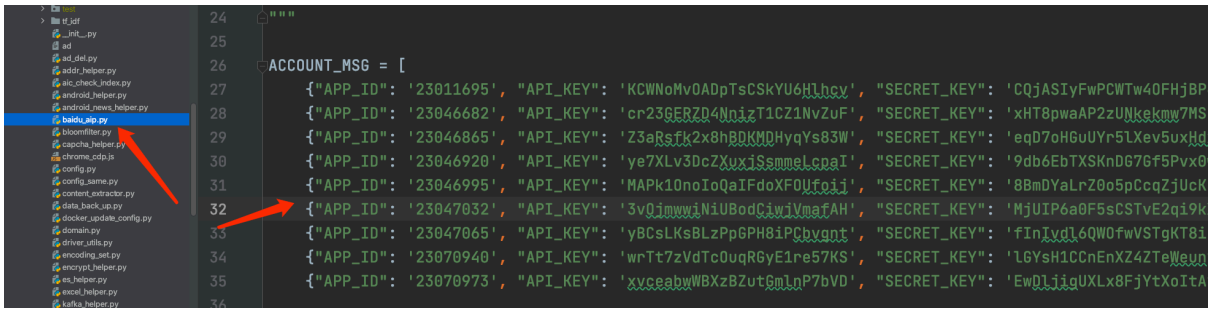
BidMainbodyContext用于招投标正文提取，主要使用一些自制的针对招投标表述的四种结构进行的泛用模型。准确率约为75%单个。测试他们的用例也在项目中，路径为html_doc/bid，用例有HTML和其答案。可用mainbody_context_impl下的__main__方法测试其准确性。



图示：招投标的特征核心抽取算法位置

nlp_math_impl（不重要）

NLP_Parse_MathImpl用于一个测试性的算法，核心调用的是LLS的算法接口和百度的接口。LLS的算法接口已关闭无人维护，百度NLP的接口可以使用，是使用我个人资源的账号池注册了10个免费账号。量不大的测试需求可以使用。



图示：这里有大量的百度NLP-API账号，算我送给公司的吧

news_cheak_impl(重点)

CommonNewsCheak 无监督查找正文链接，核心方法是其中的re_find2。其思路主要是通过长度相同的urls，对比urls之间的不同的部分，从而决定哪些部分用正则替换，哪些部分保持原状。

举例说明：

http://abc.com/news/2020/01/03/id_123456.html

http://abc.com/news/2020/02/11/id_435456.html

两个链接，最终会生成正则：

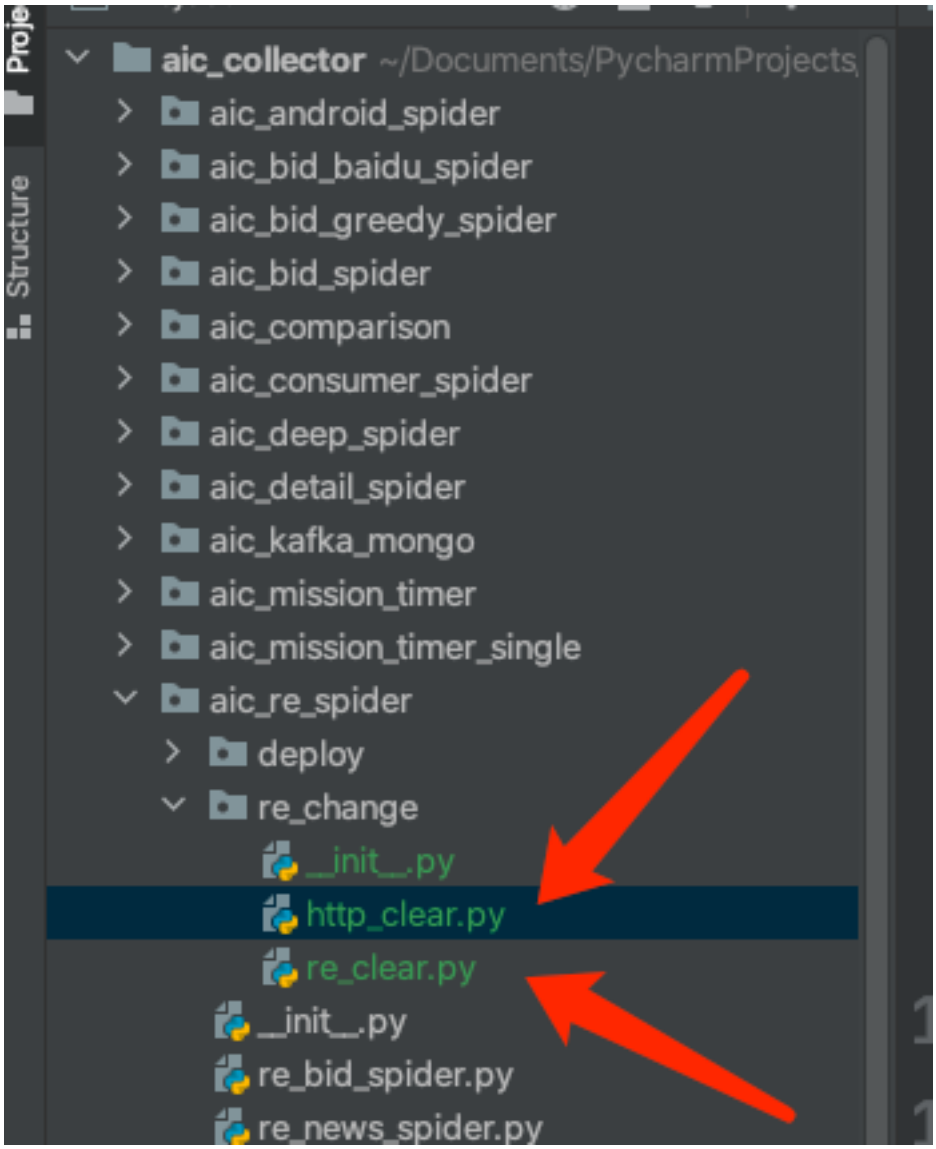
http://abc.com/news/2020/[0-9]{2}/[0-9]{2}/.{9}.html

这个正则基本对，可以识别正确的详情页。但是2020还是比较局限，因为应该有别的年份的数据。因此跑完news_cheak_impl将【监督输入】导入aic_news_range_seed表后，还有一个一次性处理部分特殊异常的代码。可以清洗seed表。

这两个代码位置为aic_re_spider.re_change，分别是http_clear.py和re_clear.py

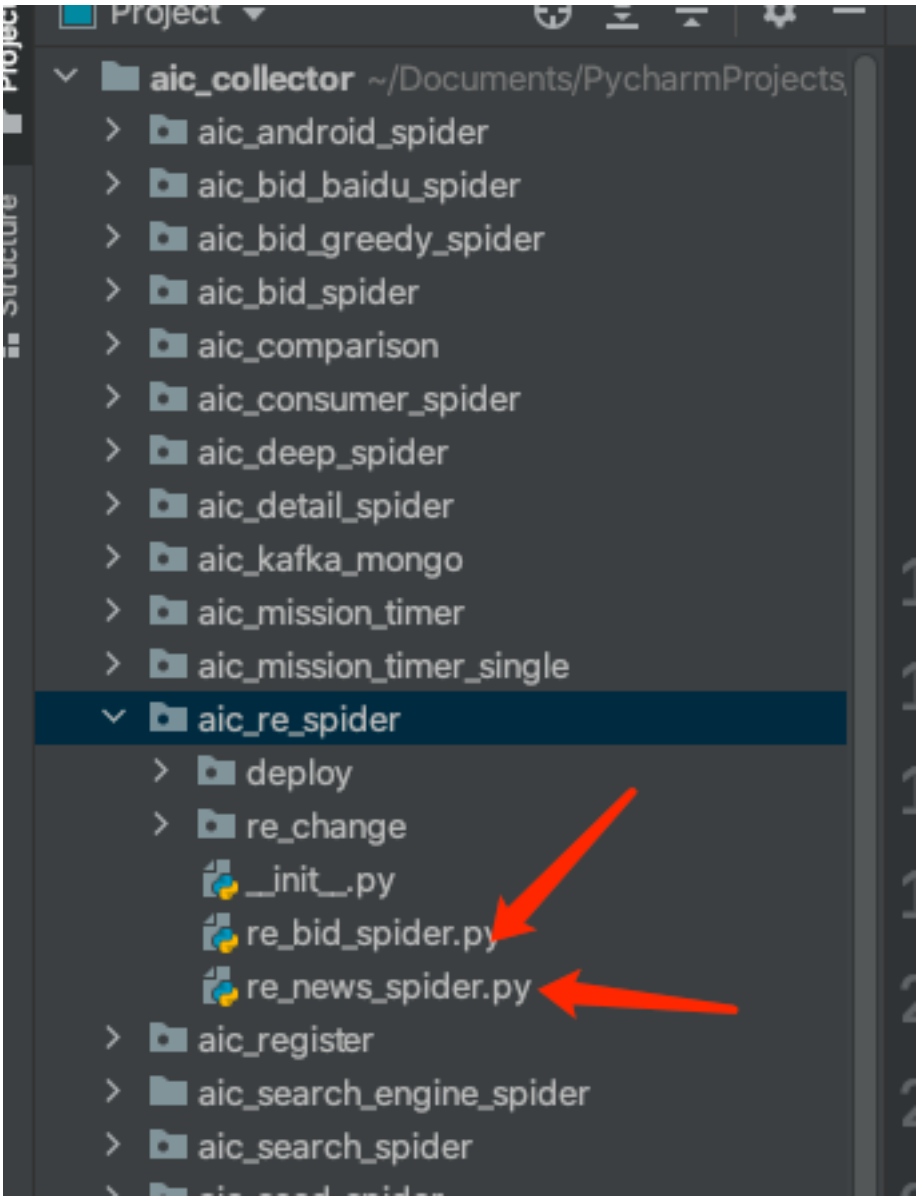
http_clear是用来对开头的http的正则清洗成(http|https),从而支持双协议。

re_clear是用来对正则中经常出错的时间进行修正，特别是年、月、日。这样即使训练数据样本数不是很多，也可以保证返回的正则准确。



图示：两个一次性清洗代码的位置，它们本地跑就可以了

SupervisedCommonNewsCheak利用CommonNewsCheak导出的【监督信息】进行识别（也就是正则）。监督信息会存到aic_news_range_seed表。该监督信息是通过aic_re_spider完成的。原则上，re_spider只需要在增加渠道后本地跑一遍就可以了。但是为了方便，aic上有re_spider服务可以使用。从而在OPS完成该操作。



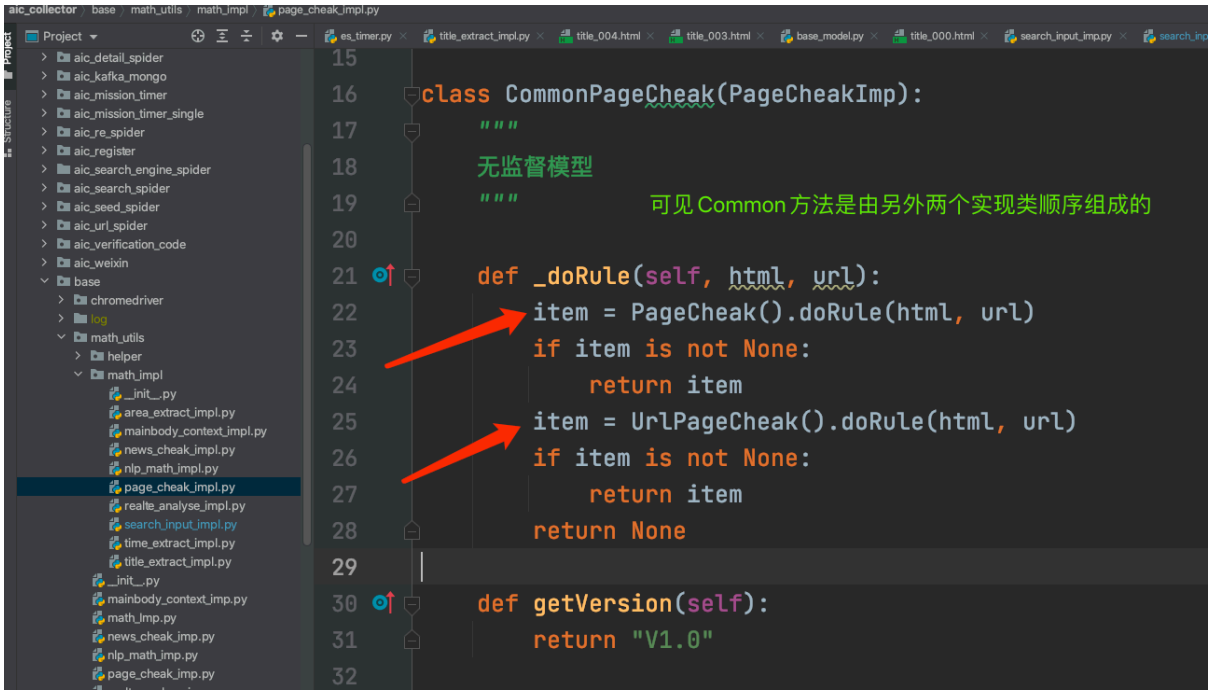
图示：两个进行【监督信息】标注的代码。在OPS上有对应服务。

BidOnclickCheck尚未使用，由刘言开发，具体功能用途请咨询刘言。

BidCommonCheck尚未使用，由刘言开发，具体功能用途请咨询刘言。

news_cheak_impl(一般重要)

CommonPageCheck通过翻页型列表查询PageCheak和Url识别型UrlPageCheak列表查询构成，详情见下面两个实现类。



图示：COMMON方法是两个CHEAK的组合使用。

PageCheak

翻页型列表查询，原理是找不重复标签中数字删除后相同的链接中最多的链接。是无监督模型。

比如HTML的A标签中存在多条类似以下链接：

http://www.baidu.com?page=0

http://www.baidu.com?page=1

……

删除数字后http://www.baidu.com?page=，是相同的，如果这个相同的在网页上出现的次数最多，他就是翻页链接，从而组装翻页url结构。

UrlPageCheak

如果翻页型找不到，他会尝试在当前URL中找存在参数num或者page的情况，那么该参数就是翻页。

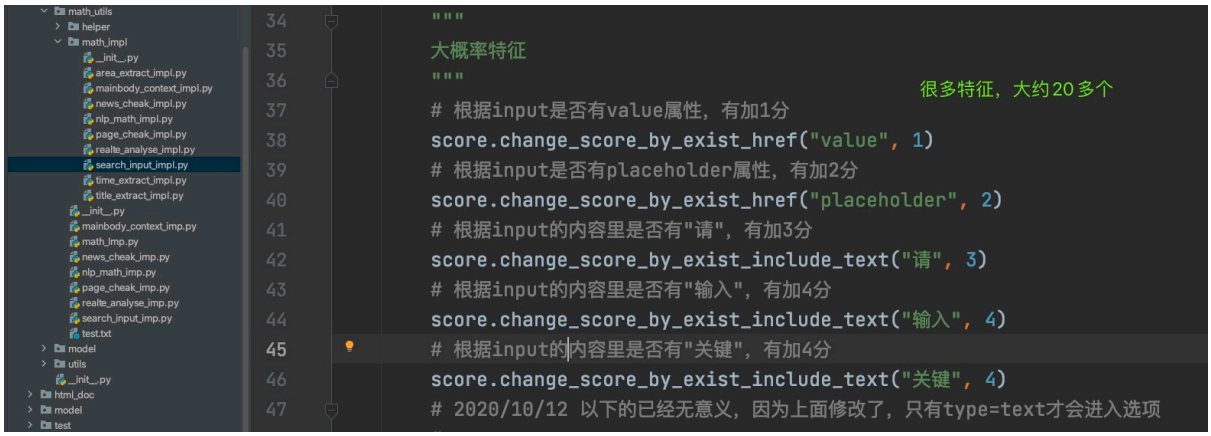
图示：参数名PAGE和NUM是写死的，并没有很复杂

realte_analyse_impl(已废弃)

该继承尝试使用主成分分析的方式进行公司相关度，最终效果不如NLP给出接口。因此废弃使用NLP给的接口，详情与范有文同学沟通。

search_input_impl(一般重要)

该继承尝试使用打分卡模型进行搜索框位置识别。具体规则已在代码中注释。



图示：有机遇打分卡的很多种规则。

time_extract_impl(一般重要)

时间抽取。主要基于正则，比较简单。最后在正文抽取impl中调用给publish_time赋值。

title_extract_impl(一般重要)

标题抽取。主要基于标签，最后在正文抽取中给title赋值。