[2023.5.12]

广东外语外贸大学数据挖掘实验室

**多语种计量指标提取工具代码使用**

# 项目启动

从项目整体结构来说，基于python flask 框架搭建 restful api。启动文件为 setup.py。项目依赖于mysql、redis。如果本地有安装docker环境的话可以通过docker-compose批量启动：

$ docker-compose -f database.yml up -d

（database.yml在项目根目录下可以找到，具体端口、用户名、密码可自行配置）

如果没有docker环境的话可以本地启动mysql、redis服务，对应修改下config目录下的配置项即可。

开发项目使用的是python3.7，因此最好使用Anaconda安装相应python版本。根目录下有一个requirements.txt文件，包括项目依赖的pypi包，对应安装即可。

$ python install -r requirements.txt

准备工作完成后可在终端输入以下代码启动整个项目：

$ set FLASK\_APP=setup

$ set FLASK\_DEBUG=true

$ flask run --host=0.0.0.0 --port=5000

# 目录结构

主要目录包括config、resources、utils、static四个目录，以下将从四个目录包含主要文件作说明：

## Config

config 目录下主要是一些工具的配置文件。包括 sqlalchemy、redis、celery、flask-mail、flask-apscheduler。config.yml 这个文件包含了所有工具的配置文件(区分开发环境和生产环境)。

## Resources

resources 目录下是整个项目的 api 所在，即相应的接口。基本上每个目录都包含两个文件 —— \_init\_.py， views.py 。views.py 是接口的主要逻辑所在，\_init\_.py 用于注册views.py提供的接口，逻辑如图 1所示。下面将逐一介绍各个目录对应的接口详情。

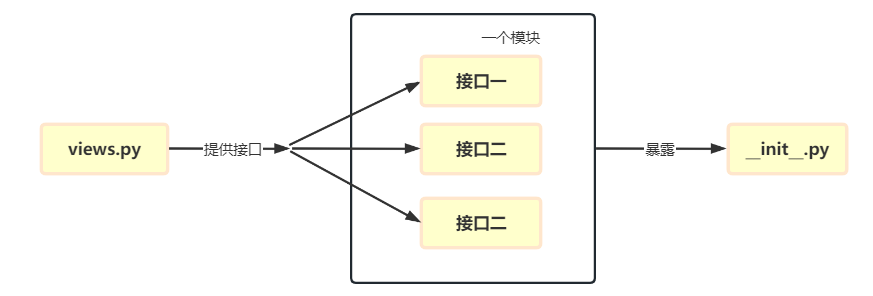


图 1 views.py与\_\_init\_\_.py逻辑关系

### common\_indicator

该目录主要提供通用指标提取接口。

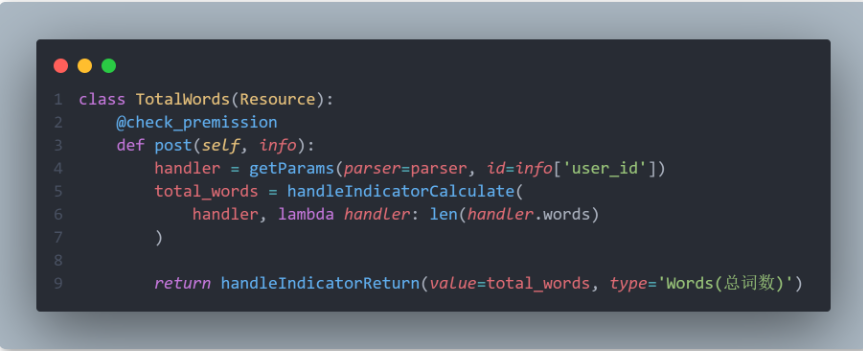


图 2 总词数提取接口

如图 2即为一个接口，其他以此类推。目前对于通用指标采用了 redis hash 缓存的策略——对处理文本生成 md5 hash，以 md5 hash 作为键值，文本处理结果作为 value 存入 redis。这样的好处是在实际提取一个文本不同指标的时候不需要重新对文本进行处理。具体的实现逻辑在 getParams 函数。

common\_indicator 里面有一个 util 目录，目录下包括一个指标提取类 CommonIndicatorHandler —— 包含所有指标的计算函数。

### en\_readability

该目录主要提供西方语种文本可读性提取接口，views.py 总体结构与 common\_indicator 类似。对应也同样存在一个 util 目录包含可读性指标公式计算类。

### Multitask

该目录主要处理任务中心、多文档上传模块，包含任务获取、任务详情、任务下载等接口。任务即在多文档上传模块新建的处理。

### open\_interace

该目录主要提供开放 api 接口（方便定制场景应用）。

### zh\_readbility

该目录主要提供中文文本可读性接口。

* feature\_bean：对应特征值提取实现函数
* get\_features：特征提取函数（调用 feature\_bean）
* method：对应方法处理类
* read\_write\_txt：得到特征提取依赖的数据项
* data\_loader.py 是数据初始化模块，挂载文本处理结果和对应方法处理类

## Utils

utils 目录主要包含各个语种的文本处理函数，useForChineseInstance 即为中文文本处理函数，其余以此类推，具体逻辑组织如图 3 所示。

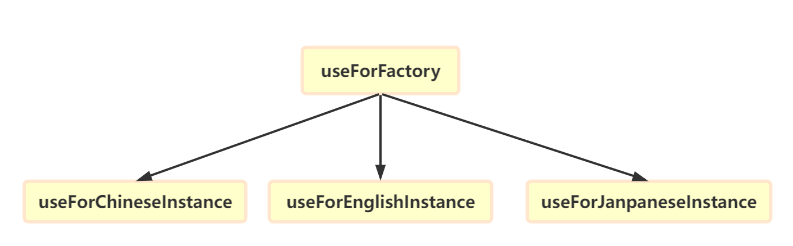


图 3 utils目录结构图示

## Static

static目录下主要是系统中需要用到的一些静态文件，包括ISO-639简称映射，pyltp模型文件等等。

# 项目部署

项目整体基于docker-compose部署，具体部署配置在根目录下docker-compose.yml。