功能书

项目分为两个模块，爬取历史文章记录和定时爬取新的文章

参考文章：<https://zhuanlan.zhihu.com/p/24302048>

具体操作：

首先采用安卓模拟器和基于nodejs的anyproxy代理服务器以及django框架来获取历史文章的URL，需要安装nodejs和anyproxy。

具体操作是命令行中输入anyproxy -i (这里的-i是支持https)，进入weixin文件夹的一级目录下，打开新的命令行输入python manage.py runserver，再模拟器上打开微信（需要设置安卓模拟器的代理服务器地址端口，与anyproxy一致），点击公众号的历史文章页面，代理服务器会向页面中插入js脚本（需要修改rule\_default.js），自动获得所有的文章数据，并交给django解析获得文章的URL。

接着通过Spider里的Crawler和getCompany来获取所需要的内容。

具体操作为先运行Crawler.py,全部下载完后运行GetCompany.py。到此为止下载完了所有的历史文章，接着运行Timed.py来实现定时爬取。

整个历史文章的消息结构为：

"list": [ //最外层的键名；只出现一次，所有内容都被它包含。

{//这个大阔号之内是一条多图文或单图文消息，通俗的说就是一天的群发都在这里

"app\_msg\_ext\_info":{//图文消息的扩展信息

"content\_url": "图文消息的链接地址",

"cover": "封面图片",

"digest": "摘要",

"is\_multi": "是否多图文，值为1和0",

"multi\_app\_msg\_item\_list": [//这里面包含的是从第二条开始的图文消息，如果is\_multi=0，这里将为空

{

"content\_url": "图文消息的链接地址",

"cover": "封面图片",

"digest": ""摘要"",

"source\_url": "阅读原文的地址",

"title": "子内容标题"

},

...//循环被省略

],

"source\_url": "阅读原文的地址",

"title": "头条标题"

},

"comm\_msg\_info":{//图文消息的基本信息

"datetime": '发布时间，值为unix时间戳',

"type": 49 //类型为49的时候是图文消息

}

},

...//循环被省略]

Spider模块

Crawler.py：

功能：主要功能为从数据库里获得文章URL，并爬取文章关键信息

类 ：crawl

函数：crawl.get\_url 从数据库中取出并返回历史文章的URL

crawl.download 根据URL下载html，并返回BeautifulSoup包装后的对象

crawl.dispose 通过BeautifulSoup解析html页面以及字符串查找替换来获得并返回我们所需要的内容

crawl.save 将内容插入数据库

GetCompany.py：

功能：从标题中提取公司名称，更新数据库（存在两种形式，一种有股票代码，一种没股票代码）

类：getCompany

函数：getCompany.get\_titles 从数据库中获得并返回所有的标题

getCompant.title\_split 对文章标题以股票代码右括号进行分割，并返回分割后的标题列表

getCompany.extract\_one 对一个分割后的标题进行正则匹配，获得并返回公司名称

getCompany.extract\_more将getCompant.title\_split和getCompany.extract\_one组合返回公司名称

getCompany.save 将内容更新到数据库

getCompany.get\_company 获取并返回所有没有股票代码的文章标题

getCompany.update\_company 将没有股票代码的标题与数据库中已有的公司名做匹配，得到未发现的公司名称，并更新。

Timed.py:

功能：定时更新文章

类：Timed

函数：Timed.get\_public\_num\_url 获得公众号的地址

Timed.get\_article\_url 获得文章的url

Timed.crawl 调用了Crawl类和GetCompany类来爬取所需内容

Timed.save 将内容插入数据库