



Python 程序设计课程

大 作 业

项目名称： 微信公众平台实现英语学习

完 成 人： 胡金辉

完成日期： 2018 年 4 月 30 日

指导教师： 赵英良

2018

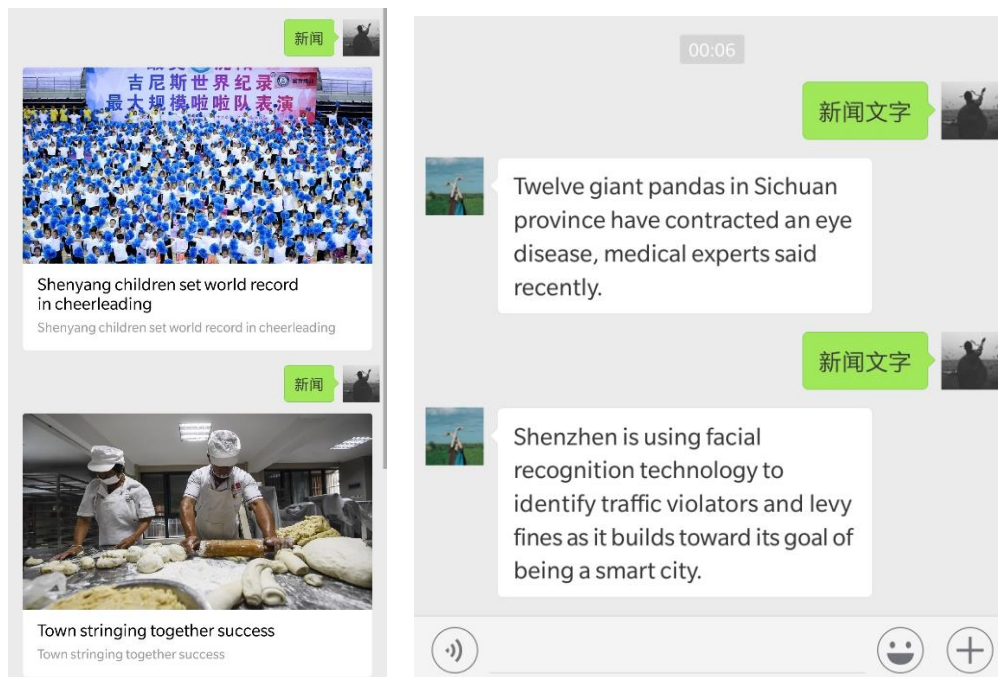
目 录

1.软件功能	3
1.1 获取当前最新英文新闻	3
1.2 托福单词的获取	4
2.软件的运行环境	4
2.1 运行平台	4
2.2 Python 版本及所需第三方库	4
3.环境搭建说明	5
3.1 配置微信公众平台	5
3.2 配置新浪云 SAE 平台	5
4.软件的设计说明	9
5.软件使用说明	17
5.1 获取当前最新英文新闻	17
5.2 托福单词的获取	19
5.3 其他信息的处理	19
6.参考资料	19

1.软件功能

1.1 获取当前最新英文新闻

回复【新闻】或【新闻文字】便可以获取当前 Chinadaily 上最新的新闻，可以多次回复获取不同的新闻。示例图如下：（具体使用方法见第五节）

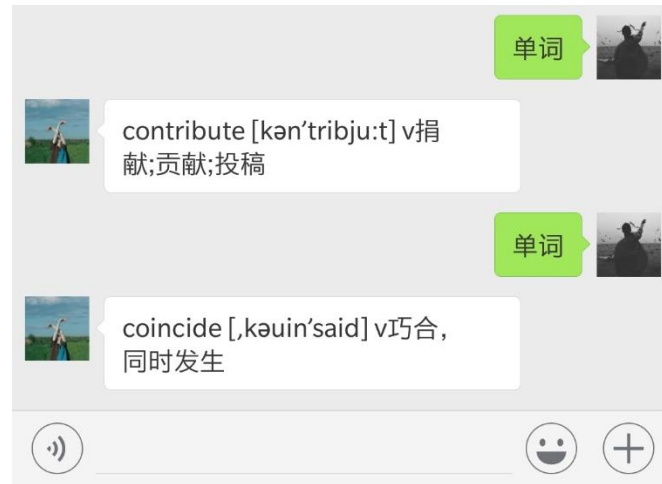


其中点开图文信息，我们会链接到该条新闻的页面。



1.2 托福单词的获取

在后台回复【单词】便可获取一个高频托福单词。（具体使用方法见第五节）



单词附带释义及音标。

2. 软件的运行环境

2.1 运行平台

1. 微信公众平台: <http://mp.weixin.qq.com>

2. 新浪云 SAE 平台: <http://sae.sina.com.cn/>

2.2 Python 版本及所需第三方库

1. Python 版本: Python 2.7

2. 第三方库:

（版本信息具体请看下文所列出的 config.yaml 文件，内有详细的版本信息。）

web.py lxml urllib2/3 requests certifi chardet idna

安装可以直接用 pip 在线安装。（例如：python -m pip install web.py）

需要上传到新浪云的可在相应目录下压缩上传。

3.环境搭建说明

3.1 配置微信公众平台

1. 你需要先在微信的公众平台注册账号，微信平台注册的时候需要你拍张手持身份证的照片，还有几天的审核期。

2. 等待微信公众审核通过后，登录公众平台后，点击高级功能。将会看到需要提供一个接入信息：



服务器配置(已启用)		修改配置	停用
服务器地址(URL)	http://[redacted].applinzi.com/weixin		
令牌(Token)	[redacted]		
消息加解密密钥(?) (EncodingAESKey)	[redacted]		
消息加解密方式	兼容模式		

点击“修改配置”进行配置。服务器地址填写新浪云服务器地址（下文提到），令牌自行设定，信息加解密密钥可以随机生成，消息加解密方式选择“兼容模式”。启用配置即可。

但如果新浪云为配置完成，无法启用配置，请先配置完成新浪云。

3.2 配置新浪云 SAE 平台

1.注册登录，在 SAE 上搭建 python 的应用，在下图的应用里选择 python 应用。

返回控制台

自定义

云市场NEW一键安装

部署环境

开发语言

PHP

Python

Java

NodeJS

Go

自定义NEW

运行环境

独享环境

共享环境

语言版本

Python2.7

代码管理

GIT

SVN

代码管理方式在创建应用成功后不可更改。

应用信息

二级域名

二级域名

.applinzi.com

仅允许由数字，字母组成，长度为4到18位。
唯一标识，也是二级域名前缀，创建后不可修改。

应用名称

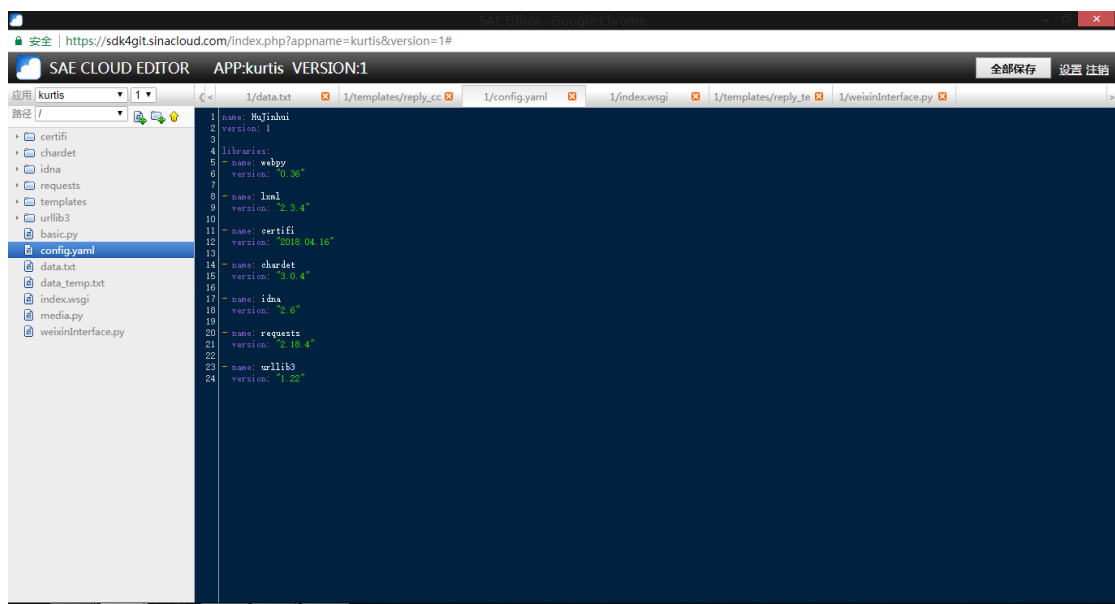
应用名称

应用的中文名称，供显示用。

填好二级域名和应用名称等，选择好语言。这里我们使用 Python 开发选择 web 应用。

创建好应用之后，在代码管理中创建一个新的版本。而后我们可以选择编辑代码。能够实现在线编辑。可以使用 git 同步本地代码。

在线编辑平台：



2.编写 config.yaml

```
1. name: HuJinhui
2. version: 1
3.
4. libraries:
5. - name: webpy
6.   version: "0.36"
7.
8. - name: lxml
9.   version: "2.3.4"
10.
11. - name: certifi
12.   version: "2018.04.16"
13.
14. - name: chardet
15.   version: "3.0.4"
16.
17. - name: idna
18.   version: "2.6"
19.
20. - name: requests
21.   version: "2.18.4"
22.
23. - name: urllib3
24.   version: "1.22"
```

其中要将你所有需要的第三方库编入其中，version 要填写正确。urllib2 平台自带不

需安装。urllib3 是 requests 库（version: 2.18.4）的运行必须的库。

3. 编写 index.wsgi:

```
1. # coding: UTF-8
2. import os
3. import sys
4. import sae
5. import web
6.
7. from weixinInterface import WeixinInterface
8.
9. urls = (
10. '/weixin', 'WeixinInterface'
11. )
12.
13. app_root = os.path.dirname(__file__)
14. #安装第三方库
15. sys.path.insert(0, os.path.join(app_root, 'requests.zip'))
16. sys.path.insert(0, os.path.join(app_root, 'urllib3.zip'))
17. sys.path.insert(0, os.path.join(app_root, 'chardet.zip'))
18. sys.path.insert(0, os.path.join(app_root, 'certifi.zip'))
19. sys.path.insert(0, os.path.join(app_root, 'idna.zip'))
20.
21. #获取 xml 所在目录
22. templates_root = os.path.join(app_root, 'templates')
23. render = web.template.render(templates_root)
24.
25. app = web.application(urls, globals()).wsgifunc()
26. application = sae.create_wsgi_app(app)
```

4.库的安装:

先将需要安装的包压缩为 zip 文件（请自行下载），并上传到根目录下。修改

index.wsgi 添加

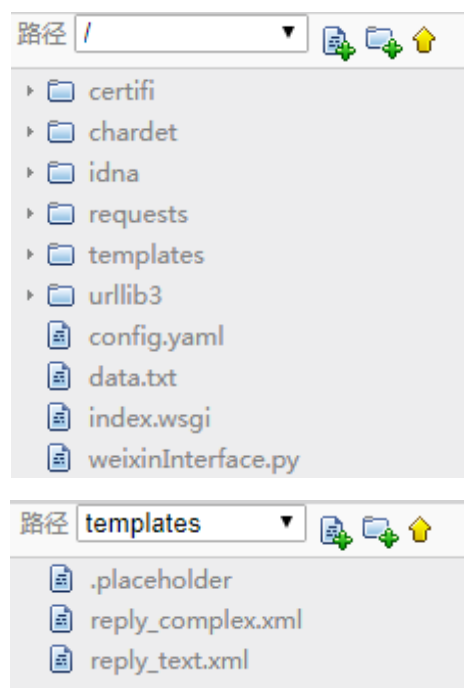
```
1. import sys
2.
3. app_root = os.path.dirname(__file__)
4. sys.path.insert(0, os.path.join(app_root, 'requests.zip'))
```


(以 requests 包为例)，到此便安装完成。

至此支持环境配置完成，将完成的代码上传到新浪云上即可。

4.软件的设计说明

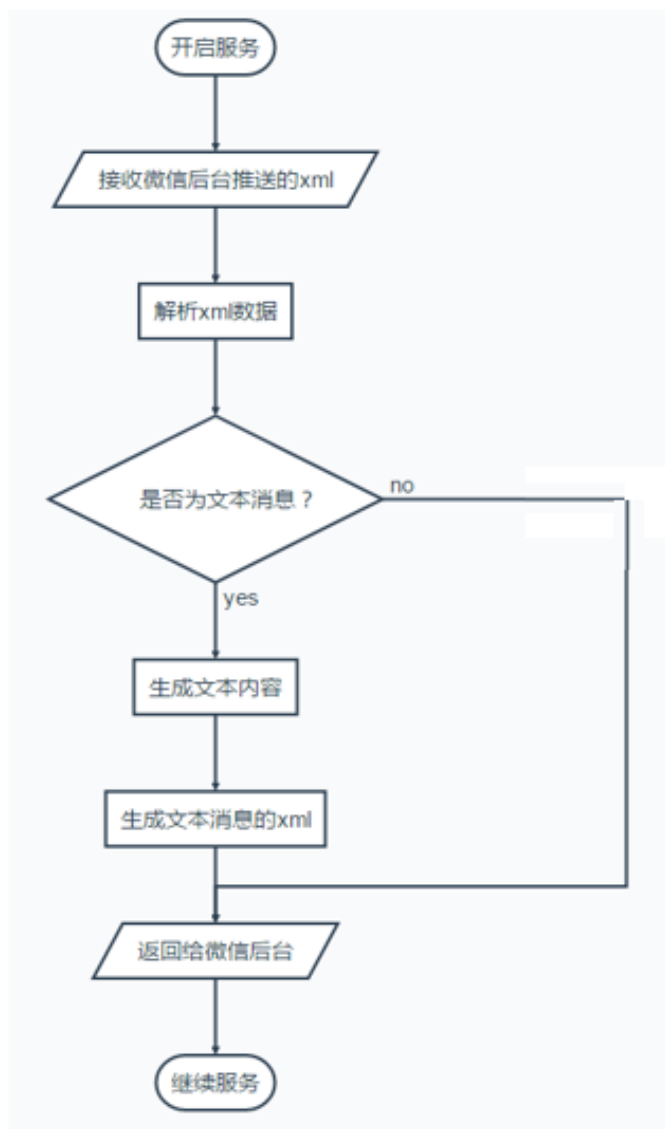
新浪云上文件目录见下：



其中 certifi、chardet、idna、requests、urllib3 均为自行上传的第三方库，上传方式见上，内容不再赘述。

总体代码总量在 500 行左右。

此程序整体的逻辑流程为：



index.wsgi 文件代码[见上](#)。主要目的是安装第三方库和完成与微信公平平台的联通。

config.yaml 文件代码[见上](#)。主要是声明配置。

data.txt 文件内包含内容为托福单词，文件部分内容见图，在此不再赘述：

```

1 1. abandon [ə' bændən] v. 放弃;抛弃
2
3 2. abstract ['æbstrækt, æb'strækt] a. 抽象的 n. 摘要
4
5 3. academy [ə' kædəmi] n. 研究院;学院;学会
6
7 4. access ['ækses, 'ækses, æk'ses] n. 接近;入口
8
9 5. accommodate [ə' kɒmədeɪt] v. 容纳;给...提供住宿;使适应
10
11 6. accompany [ə' kʌmpəni] v. 陪伴;伴随
12

```

程序的主要逻辑实现在 weixinInterface.py 文件，代码如下：

```
1. # -*- coding: utf-8
2. #调用第三方库
3. import hashlib
4. import web
5. import lxml
6. import time
7. import os
8. import urllib2,json
9. from lxml import etree
10. import random
11. import requests
12. import re
13.
14. #建立类实现功能
15. class WeixinInterface:
16.
17.     #初始化类
18.     def __init__(self):
19.         self.app_root = os.path.dirname(__file__)
20.         #这是 xml 文件的目录
21.         self.templates_root = os.path.join(self.app_root, 'templates')
22.         self.render = web.template.render(self.templates_root)
23.
24.     def GET(self):
25.         #获取输入参数
26.         data = web.input()
27.         signature=data.signature
28.         timestamp=data.timestamp
29.         nonce=data.nonce
30.         echostr=data.echostr
31.
32.         #自己的 token，处于隐私做了处理
33.         token="*****"
34.
35.         #字典序排序
36.         list=[token,timestamp,nonce]
37.         list.sort()
38.
39.         sha1=hashlib.sha1()
40.         map(sha1.update,list)
41.         hashcode=sha1.hexdigest()
42.         #sha1 加密算法
43.
44.         #如果是来自微信的请求，则回复 echostr
```

```

45.         if hashcode == signature:
46.             return echostr
47.
48.     #报文回复
49.     def POST(self):
50.         #获得 post 来的数据
51.         str_xml = web.data()
52.         #进行 XML 解析
53.         xml = etree.fromstring(str_xml)
54.         #获得用户所输入的内容
55.         content=xml.find("Content").text
56.         msgType=xml.find("MsgType").text
57.         #获得发送者 ID
58.         fromUser=xml.find("FromUserName").text
59.         #获得接受者 ID
60.         toUser=xml.find("ToUserName").text
61.
62.         #对所获内容进行分类处理
63.         #使用 if 语句而不使用 if...else 语句的原因是新浪云对于 else 的支持有问题，会
        偶发报错（待进一步判断）
64.         if(u'新闻' in content):
65.             if(u'文字' in content):
66.                 try:
67.                     news,pics,title,new_url=self.getNews()
68.                     news[0]=news[0].replace("\'", "'")
69.                     #回复标准 xml 信息，这个为文字 xml
70.                     return self.render.reply_text(fromUser,toUser,int(time.t
ime()),news[0])
71.                 except:
72.                     cont=u'er101'
73.                     #出错报错并返回错误代码
74.                     return self.render.reply_text(fromUser,toUser,int(time.t
ime()),u"服务器开小差了:( \n 请重新回复关键字! \n"+cont)
75.             if(u'文字' not in content):
76.                 try:
77.                     #回复标准 xml 信息，这个为图文信息 xml
78.                     news,pics,title,new_url=self.getNews()
79.                     title=title.replace("\'", "'")
80.                     return self.render.reply_complex(fromUser,toUser,int(tim
e.time()),title,title,pics,new_url)
81.                 except:
82.                     #出错报错并返回错误代码
83.                     cont=u'er102'

```

```

84.             return self.render.reply_text(fromUser,toUser,int(time.time()),u"服务器开小差了:( \n 请重新回复关键字! \n"+cont)
85.
86.         if(u'单词' in content):
87.             try:
88.                 word=str(self.words())
89.                 word=word.strip()
90.                 word=word.replace("\'","''")
91.                 #回复标准 xml 信息, 这个为文字 xml
92.                 return self.render.reply_text(fromUser,toUser,int(time.time()),word)
93.             except:
94.                 cont=u'error103'
95.                 #回复标准 xml 信息, 这个为文字 xml
96.                 return self.render.reply_text(fromUser,toUser,int(time.time()),u"服务器开小差了:( \n 请重新回复关键字! \n"+cont)
97.
98.         if(u'单词' not in content) and (u'新闻' not in content):
99.             #回复标准 xml 信息, 这个为文字 xml
100.            #作为提示信息出现, 保证每次输入均有回复
101.            cont=u'回复【新闻】或【文字新闻】获取最新英文新闻。 \n 回复【单词】获取托福单词\n'
102.            return self.render.reply_text(fromUser,toUser,int(time.time()),cont)
103.
104.        #获取网页
105.        def getWebpage(self,url,kw):
106.
107.            headers={
108.                'User-Agent':'Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36(KHTML,like Gecko) Chrome/57.0.2987.110 Safari/573.36','referer':'#
109.            }
110.
111.            try:
112.                r=requests.get(url,headers=headers,\
113.                               params=kw,timeout=30)
114.                #print(r.status_code)
115.                r.raise_for_status()
116.                r.encoding=r.apparent_encoding
117.                return r.text
118.            except:
119.                return 'error'
120.
121.        #获取新闻

```

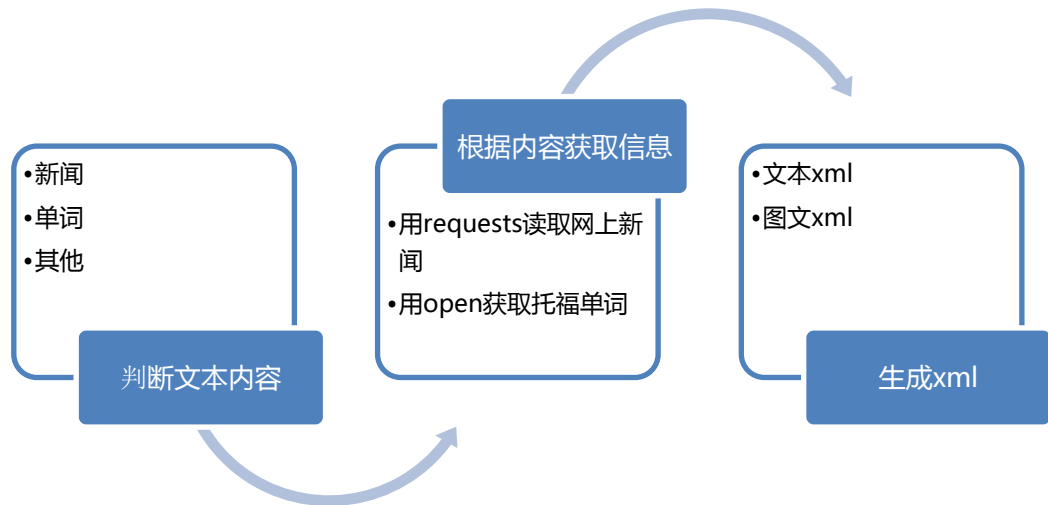
```

122.     def getNews(self):
123.         kw={'wd':'python requests'}
124.         url=['http://www.chinadaily.com.cn/china']
125.         #调用 getWebpage 函数下载网页 html 文件
126.         s=self.getWebpage(url[0],kw)
127.         #使用正则表达式提取目标新闻地址
128.         match=re.findall('<a target="_top" shape="rect" href=.{+?}</a>',s)
129.         i=0
130.         res=[]
131.         res_url=[]
132.         res_name=[]
133.         for x in match:
134.             if i>20:
135.                 #运用总结规则筛选有效地址
136.                 if ('img' not in x)and('font' not in x)and('span' not in x)
137.                 :
138.                     res.append(str(x))
139.             i+=1
140.             #设置结束条件
141.             if i ==150:
142.                 break
143.
144.         #处理字符串并将超链接与文章标题保存
145.         for x in res:
146.             x=x.split('\n')
147.             num=x[6].find('</a>')
148.             if len(x[6])>25:
149.                 res_url.append(x[5])
150.                 res_name.append(x[6][1:num])
151.
152.         #将保存下来的文档随机抽取一个进行后续信息收集
153.         num=random.randint(0,len(res_url)-1)
154.         s=self.getWebpage(res_url[num],kw)
155.         title=res_name[num]
156.         new_url=res_url[num]
157.
158.         #收集新闻内容
159.         match=re.findall('<p>.{+?}</p>',s)#正则表达式一
160.         news=[]
161.         for x in match:
162.             if ('<a>' not in x):
163.                 x=x[3:]
164.                 x=x.replace('</p>','\n')

```

```
165.         news.append(x)
166.
167.         match=re.findall('<figcaption.+?>/figcaption>',s)#正则表达式二
168.         for x in match:
169.             if ('<a>' not in x):
170.                 #print(x)
171.                 x.replace('<', '《')
172.                 x.replace('>', '》')
173.                 x.replace('\n', '')
174.                 x=x.split('》')
175.                 x=x[1]
176.                 x=x.replace('</figcaption>', '\n')
177.                 news.append(x)
178.
179.         #获取新闻中图片地址
180.         match=re.findall('<img src=.*?data-mimetype',s)
181.         for x in match:
182.             x=x.split('\n')
183.             pics=x[1]
184.
185.         return news,pics,title,new_url
186.
187.     #读取 data.txt 文件获取托福单词
188.     def words(self):
189.         f=open('data.txt','r')
190.         data=f.readlines()
191.         num=f.tell()
192.         #随机选取一个
193.         num=random.randint(0,num-100)
194.         f.seek(num)
195.
196.         data=f.readline()
197.         data=f.readline()
198.         if data=='\n':
199.             data=f.readline()
200.         f.close()
201.
202.         data=data.split('.')
203.         data=data[1:]
204.         data=''.join(data)
205.         return data
```

它生成文本信息的 xml 逻辑顺序为:



templates 文件夹下主要配置的是 xml 信息:

reply_complex.xml:

```
1. $def with (toUser,fromUser,createTime,title,description,picur,url)
2.
3.
4.
5. <xml>
6. <ToUserName><![CDATA[$toUser]]></ToUserName>
7. <FromUserName><![CDATA[$fromUser]]></FromUserName>
8. <CreateTime>$createTime</CreateTime>
9. <MsgType><![CDATA[news]]></MsgType>
10. <ArticleCount>1</ArticleCount>
11. <Articles>
12.   <item>
13.     <Title><![CDATA[$title]]></Title>
14.     <Description><![CDATA[$description]]></Description>
15.     <PicUrl><![CDATA[$picur]]></PicUrl>
16.     <Url><![CDATA[$url]]></Url>
17.   </item>
18. </Articles>
19. </xml>
```

reply_text.xml:

```
1. $def with (toUser,fromUser,createTime,content)
2.
```



```
3. <xml>
4. <ToUserName><![CDATA[$toUser]]></ToUserName>
5. <FromUserName><![CDATA[$fromUser]]></FromUserName>
6. <CreateTime>$createTime</CreateTime>
7. <MsgType><![CDATA[text]]></MsgType>
8. <Content><![CDATA[$content]]></Content>
9. </xml>
```

此为微信公众平台所用格式。

该代码运用了所学的几乎所有 python 基础知识，主要侧重文件的读取、第三方库（如 requests）的调用、正则表达式等，还灵活运用了其他一些知识，如 web 前端，服务器运维等知识，真正做到了知识的灵活运用。

本程序的创新之处在于不同于以往的单机程序，此程序可以在线访问，是一个有真正使用途径以及可以长时间使用的程序，本次大作业不只是一份作业，还可以帮助同学复习托福考试，增长英语水平。

5.软件使用说明

5.1 获取当前最新英文新闻

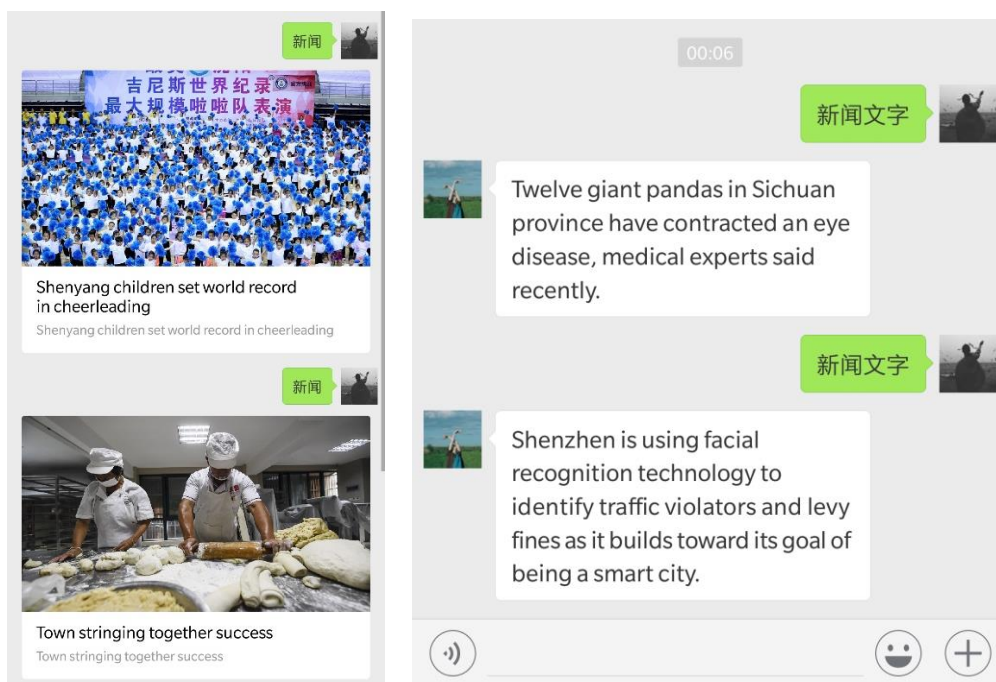
【注意：本服务器只有 10 天有效期，学生认证还在通过中不能续费，如遇“微信公众号暂时不能提供服务”错误请联系我给服务器续费！】

首先我们通过关注“单柯成森”微信公众号可以在线实现这个功能，微信公众号的二维码见下：



扫描关注后回复【新闻】或【新闻文字】便可以获取当前 Chinadaily 上最新的新闻，可以多次回复获取不同的新闻。示例图如下：

（限于微信公众平台回复文字的要求，新闻文字只能回复第一段）

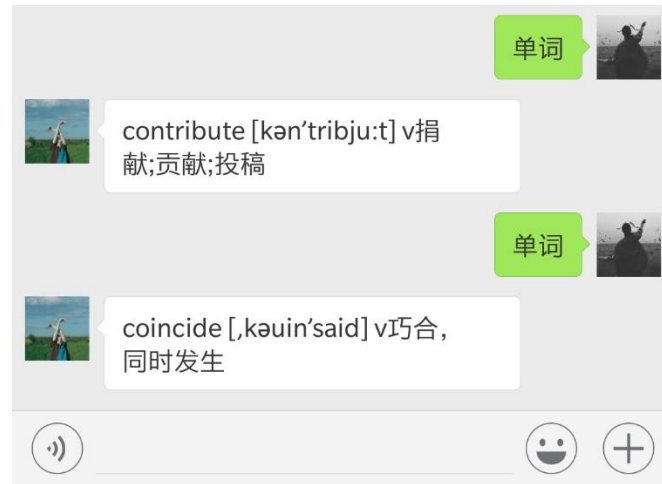


其中点开图文信息，我们会链接到该条新闻的页面。



5.2 托福单词的获取

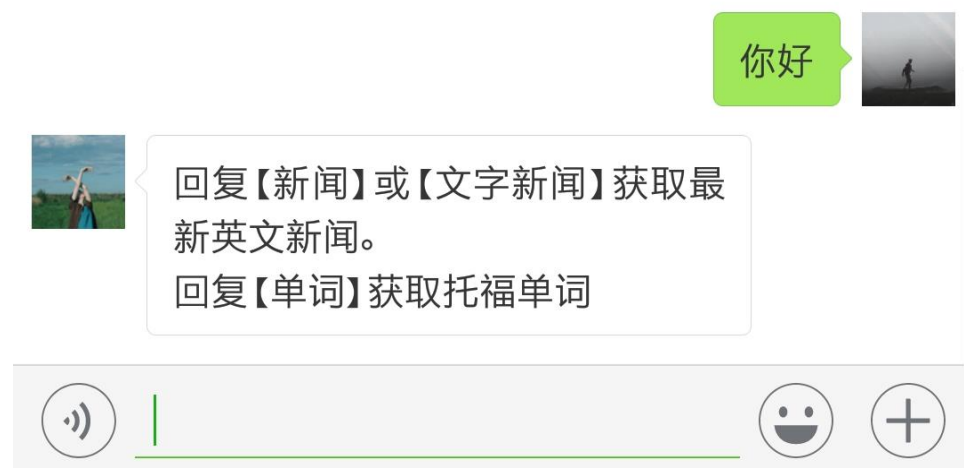
在后台回复【单词】便可获取一个高频托福单词。



单词附带释义及音标。

5.3 其他信息的处理

如遇其它信息，则会回复提示信息。



6.参考资料:

https://mp.weixin.qq.com/wiki?t=resource/res_main&id=mp1421140543

https://mp.weixin.qq.com/wiki?t=resource/res_main&id=mp1472017492_58YV5

<https://blog.csdn.net/kobe2016/article/details/79009023>