**计算机科学与工程学院 实验报告**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **实验课程名称** | | **Python编程与数据分析** | | |
| **专业** | **计算机科学与技术专业** | | **班级** |  |
| **学号** |  | | **姓名** |  |
| **实验心得与问题** | | | | |
| **实验报告正文**  **实验操作系统及环境：**  **操作系统：**  Linux 5.10.0-1-amd64 #1 SMP Debian 5.10.4-1 (2020-12-31) x86\_64 GNU/Linux  **Python环境：**  Python 3.9.1+ (default, Jan 10 2021, 15:42:50)   [GCC 10.2.1 20201224] on linux  **实验题目 第一个：**  **制作一个天气预报应用程序：输入城市名称，给出当天天气情况，并用图文并茂形式展示出来。**  **对应源代码：**  weather\_com\_cn 为爬取www.weather.com.cn 网页的模块，通过分析使用re模块爬取输入城市的预选城市，获得js文件，再通过解析js文件得到城市的code列表，映射到GUI的列表中，通过列表选择城市，将天气的情况显示在右侧。  GUI通过PyQt5和pyqt5\_tools提供，通过可视化生成界面和py文件，然后修改槽和信号将其映射到自定义函数上从而实现按下按钮的响应。  weather\_com\_cn.py  def search(city):  '''  input: city name  output: a dict include the cities and the code  '''  import requests  import json  url = r'http://toy1.weather.com.cn/search?cityname=' + city + r'&callback=success\_jsonpCallback&\_=1610587632208'  res = requests.get(url)  res.encoding = 'utf-8'  result0 = json.loads(res.text[22:-1])  result1 = {}  for item in result0:  content = item['ref'].split('~')  result1[content[-1] + content[2]] = content[0]  #print(result1)  return result1  def get\_weather(city\_code):  '''  input: city code  output: tempurature, wind ...  '''  from bs4 import BeautifulSoup  import requests  url = r'http://www.weather.com.cn/weather1d/' + str(city\_code) + r'.shtml'  headers = {  'User-Agent':'Mozilla/5.0 (Windows NT 10.0; Win64; x64) '  'AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.102 Safari/537.36'  }  re = requests.get(url, headers = headers)  html = re.content.decode('UTF-8')  bs\_obj = BeautifulSoup(html, "html.parser")  bs\_obj = bs\_obj.find('div',class\_ = 't')  h = bs\_obj.find\_all('h1')  p = bs\_obj.find\_all('p')  big = bs\_obj.find\_all('big')  info = []  daytime = {}  daytime['time'] = h[0].text  daytime['wea'] = p[0].text  daytime['weap'] = big[0].get('class')[1]  daytime['tem'] = p[1].span.text + p[1].em.text  daytime['win'] = p[2].span.get('title') + p[2].span.text  daytime['winp'] = p[2].i.get('class')[0]  daytime['suntime'] = p[3].span.text  info.append(daytime)  night = {}  night['time'] = h[1].text  night['wea'] = p[4].text  night['weap'] = big[1].get('class')[1]  night['tem'] = p[5].span.text + p[1].em.text  night['win'] = p[6].span.get('title') + p[2].span.text  night['winp'] = p[6].i.get('class')[0]  night['suntime'] = p[7].span.text  info.append(night)  return info  def debug(city\_code):  pass  weather.py  # -\*- coding: utf-8 -\*-  # Form implementation generated from reading ui file 'weather.ui'  #  # Created by: PyQt5 UI code generator 5.15.2  #  # WARNING: Any manual changes made to this file will be lost when pyuic5 is  # run again. Do not edit this file unless you know what you are doing.  from PyQt5 import QtCore, QtGui, QtWidgets  class Ui\_weather(object):  def setupUi(self, weather):  weather.setObjectName("weather")  weather.resize(638, 702)  weather.setMaximumSize(QtCore.QSize(16777215, 16777215))  self.centralwidget = QtWidgets.QWidget(weather)  self.centralwidget.setObjectName("centralwidget")  self.gridLayout = QtWidgets.QGridLayout(self.centralwidget)  self.gridLayout.setObjectName("gridLayout")  self.verticalLayout\_4 = QtWidgets.QVBoxLayout()  self.verticalLayout\_4.setObjectName("verticalLayout\_4")  self.horizontalLayout\_3 = QtWidgets.QHBoxLayout()  self.horizontalLayout\_3.setObjectName("horizontalLayout\_3")  self.verticalLayout\_3 = QtWidgets.QVBoxLayout()  self.verticalLayout\_3.setObjectName("verticalLayout\_3")  self.search = QtWidgets.QPlainTextEdit(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum, QtWidgets.QSizePolicy.Maximum)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.search.sizePolicy().hasHeightForWidth())  self.search.setSizePolicy(sizePolicy)  self.search.setMinimumSize(QtCore.QSize(180, 50))  self.search.setMaximumSize(QtCore.QSize(180, 50))  self.search.setObjectName("search")  self.verticalLayout\_3.addWidget(self.search)  self.select = QtWidgets.QListWidget(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Preferred, QtWidgets.QSizePolicy.Preferred)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.select.sizePolicy().hasHeightForWidth())  self.select.setSizePolicy(sizePolicy)  self.select.setMinimumSize(QtCore.QSize(180, 350))  self.select.setMaximumSize(QtCore.QSize(180, 16777215))  self.select.setBaseSize(QtCore.QSize(0, 0))  self.select.setObjectName("select")  self.verticalLayout\_3.addWidget(self.select)  self.pushButton = QtWidgets.QPushButton(self.centralwidget)  self.pushButton.setMinimumSize(QtCore.QSize(180, 50))  self.pushButton.setMaximumSize(QtCore.QSize(180, 50))  self.pushButton.setObjectName("pushButton")  self.verticalLayout\_3.addWidget(self.pushButton)  self.horizontalLayout\_3.addLayout(self.verticalLayout\_3)  self.verticalLayout = QtWidgets.QVBoxLayout()  self.verticalLayout.setObjectName("verticalLayout")  self.time0 = QtWidgets.QLabel(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum, QtWidgets.QSizePolicy.Maximum)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.time0.sizePolicy().hasHeightForWidth())  self.time0.setSizePolicy(sizePolicy)  self.time0.setMinimumSize(QtCore.QSize(150, 50))  self.time0.setMaximumSize(QtCore.QSize(150, 50))  font = QtGui.QFont()  font.setPointSize(16)  font.setStyleStrategy(QtGui.QFont.PreferDefault)  self.time0.setFont(font)  self.time0.setText("")  self.time0.setObjectName("time0")  self.verticalLayout.addWidget(self.time0)  self.weap0 = QtWidgets.QLabel(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum, QtWidgets.QSizePolicy.Maximum)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.weap0.sizePolicy().hasHeightForWidth())  self.weap0.setSizePolicy(sizePolicy)  self.weap0.setMinimumSize(QtCore.QSize(150, 150))  self.weap0.setMaximumSize(QtCore.QSize(150, 150))  font = QtGui.QFont()  font.setPointSize(18)  font.setStyleStrategy(QtGui.QFont.PreferDefault)  self.weap0.setFont(font)  self.weap0.setText("")  self.weap0.setObjectName("weap0")  self.verticalLayout.addWidget(self.weap0)  self.wea0 = QtWidgets.QLabel(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum, QtWidgets.QSizePolicy.Maximum)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.wea0.sizePolicy().hasHeightForWidth())  self.wea0.setSizePolicy(sizePolicy)  self.wea0.setMinimumSize(QtCore.QSize(150, 25))  self.wea0.setMaximumSize(QtCore.QSize(150, 25))  font = QtGui.QFont()  font.setPointSize(18)  font.setStyleStrategy(QtGui.QFont.PreferDefault)  self.wea0.setFont(font)  self.wea0.setText("")  self.wea0.setObjectName("wea0")  self.verticalLayout.addWidget(self.wea0)  self.tem0 = QtWidgets.QLabel(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum, QtWidgets.QSizePolicy.Maximum)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.tem0.sizePolicy().hasHeightForWidth())  self.tem0.setSizePolicy(sizePolicy)  self.tem0.setMinimumSize(QtCore.QSize(150, 50))  self.tem0.setMaximumSize(QtCore.QSize(150, 50))  font = QtGui.QFont()  font.setPointSize(22)  font.setStyleStrategy(QtGui.QFont.PreferDefault)  self.tem0.setFont(font)  self.tem0.setText("")  self.tem0.setObjectName("tem0")  self.verticalLayout.addWidget(self.tem0)  self.horizontalLayout = QtWidgets.QHBoxLayout()  self.horizontalLayout.setObjectName("horizontalLayout")  self.winp0 = QtWidgets.QLabel(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum, QtWidgets.QSizePolicy.Maximum)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.winp0.sizePolicy().hasHeightForWidth())  self.winp0.setSizePolicy(sizePolicy)  self.winp0.setMinimumSize(QtCore.QSize(50, 50))  self.winp0.setMaximumSize(QtCore.QSize(50, 50))  font = QtGui.QFont()  font.setPointSize(18)  font.setStyleStrategy(QtGui.QFont.PreferDefault)  self.winp0.setFont(font)  self.winp0.setText("")  self.winp0.setObjectName("winp0")  self.horizontalLayout.addWidget(self.winp0)  self.win0 = QtWidgets.QLabel(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum, QtWidgets.QSizePolicy.Maximum)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.win0.sizePolicy().hasHeightForWidth())  self.win0.setSizePolicy(sizePolicy)  self.win0.setMinimumSize(QtCore.QSize(100, 50))  self.win0.setMaximumSize(QtCore.QSize(100, 50))  font = QtGui.QFont()  font.setPointSize(10)  font.setStyleStrategy(QtGui.QFont.PreferDefault)  self.win0.setFont(font)  self.win0.setText("")  self.win0.setObjectName("win0")  self.horizontalLayout.addWidget(self.win0)  self.verticalLayout.addLayout(self.horizontalLayout)  self.suntime0 = QtWidgets.QLabel(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum, QtWidgets.QSizePolicy.Maximum)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.suntime0.sizePolicy().hasHeightForWidth())  self.suntime0.setSizePolicy(sizePolicy)  self.suntime0.setMinimumSize(QtCore.QSize(150, 50))  self.suntime0.setMaximumSize(QtCore.QSize(150, 50))  font = QtGui.QFont()  font.setPointSize(18)  font.setStyleStrategy(QtGui.QFont.PreferDefault)  self.suntime0.setFont(font)  self.suntime0.setText("")  self.suntime0.setObjectName("suntime0")  self.verticalLayout.addWidget(self.suntime0)  self.horizontalLayout\_3.addLayout(self.verticalLayout)  self.verticalLayout\_2 = QtWidgets.QVBoxLayout()  self.verticalLayout\_2.setObjectName("verticalLayout\_2")  self.time1 = QtWidgets.QLabel(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum, QtWidgets.QSizePolicy.Maximum)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.time1.sizePolicy().hasHeightForWidth())  self.time1.setSizePolicy(sizePolicy)  self.time1.setMinimumSize(QtCore.QSize(150, 50))  self.time1.setMaximumSize(QtCore.QSize(150, 50))  font = QtGui.QFont()  font.setPointSize(16)  font.setStyleStrategy(QtGui.QFont.PreferDefault)  self.time1.setFont(font)  self.time1.setText("")  self.time1.setObjectName("time1")  self.verticalLayout\_2.addWidget(self.time1)  self.weap1 = QtWidgets.QLabel(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum, QtWidgets.QSizePolicy.Maximum)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.weap1.sizePolicy().hasHeightForWidth())  self.weap1.setSizePolicy(sizePolicy)  self.weap1.setMinimumSize(QtCore.QSize(150, 150))  self.weap1.setMaximumSize(QtCore.QSize(150, 150))  font = QtGui.QFont()  font.setPointSize(18)  font.setStyleStrategy(QtGui.QFont.PreferDefault)  self.weap1.setFont(font)  self.weap1.setText("")  self.weap1.setObjectName("weap1")  self.verticalLayout\_2.addWidget(self.weap1)  self.wea1 = QtWidgets.QLabel(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum, QtWidgets.QSizePolicy.Maximum)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.wea1.sizePolicy().hasHeightForWidth())  self.wea1.setSizePolicy(sizePolicy)  self.wea1.setMaximumSize(QtCore.QSize(150, 25))  font = QtGui.QFont()  font.setPointSize(18)  font.setStyleStrategy(QtGui.QFont.PreferDefault)  self.wea1.setFont(font)  self.wea1.setText("")  self.wea1.setObjectName("wea1")  self.verticalLayout\_2.addWidget(self.wea1)  self.tem1 = QtWidgets.QLabel(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum, QtWidgets.QSizePolicy.Maximum)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.tem1.sizePolicy().hasHeightForWidth())  self.tem1.setSizePolicy(sizePolicy)  self.tem1.setMinimumSize(QtCore.QSize(150, 50))  self.tem1.setMaximumSize(QtCore.QSize(150, 50))  font = QtGui.QFont()  font.setPointSize(22)  font.setStyleStrategy(QtGui.QFont.PreferDefault)  self.tem1.setFont(font)  self.tem1.setText("")  self.tem1.setObjectName("tem1")  self.verticalLayout\_2.addWidget(self.tem1)  self.horizontalLayout\_2 = QtWidgets.QHBoxLayout()  self.horizontalLayout\_2.setObjectName("horizontalLayout\_2")  self.winp1 = QtWidgets.QLabel(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum, QtWidgets.QSizePolicy.Maximum)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.winp1.sizePolicy().hasHeightForWidth())  self.winp1.setSizePolicy(sizePolicy)  self.winp1.setMinimumSize(QtCore.QSize(50, 50))  self.winp1.setMaximumSize(QtCore.QSize(50, 50))  font = QtGui.QFont()  font.setPointSize(18)  font.setStyleStrategy(QtGui.QFont.PreferDefault)  self.winp1.setFont(font)  self.winp1.setText("")  self.winp1.setObjectName("winp1")  self.horizontalLayout\_2.addWidget(self.winp1)  self.win1 = QtWidgets.QLabel(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum, QtWidgets.QSizePolicy.Maximum)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.win1.sizePolicy().hasHeightForWidth())  self.win1.setSizePolicy(sizePolicy)  self.win1.setMinimumSize(QtCore.QSize(100, 50))  self.win1.setMaximumSize(QtCore.QSize(100, 50))  font = QtGui.QFont()  font.setPointSize(10)  font.setStyleStrategy(QtGui.QFont.PreferDefault)  self.win1.setFont(font)  self.win1.setText("")  self.win1.setObjectName("win1")  self.horizontalLayout\_2.addWidget(self.win1)  self.verticalLayout\_2.addLayout(self.horizontalLayout\_2)  self.suntime1 = QtWidgets.QLabel(self.centralwidget)  sizePolicy = QtWidgets.QSizePolicy(QtWidgets.QSizePolicy.Maximum, QtWidgets.QSizePolicy.Maximum)  sizePolicy.setHorizontalStretch(0)  sizePolicy.setVerticalStretch(0)  sizePolicy.setHeightForWidth(self.suntime1.sizePolicy().hasHeightForWidth())  self.suntime1.setSizePolicy(sizePolicy)  self.suntime1.setMinimumSize(QtCore.QSize(150, 50))  self.suntime1.setMaximumSize(QtCore.QSize(150, 50))  font = QtGui.QFont()  font.setPointSize(18)  font.setStyleStrategy(QtGui.QFont.PreferDefault)  self.suntime1.setFont(font)  self.suntime1.setText("")  self.suntime1.setObjectName("suntime1")  self.verticalLayout\_2.addWidget(self.suntime1)  self.horizontalLayout\_3.addLayout(self.verticalLayout\_2)  self.verticalLayout\_4.addLayout(self.horizontalLayout\_3)  self.warning = QtWidgets.QLabel(self.centralwidget)  self.warning.setMinimumSize(QtCore.QSize(0, 15))  self.warning.setMaximumSize(QtCore.QSize(16777215, 15))  self.warning.setText("")  self.warning.setObjectName("warning")  self.verticalLayout\_4.addWidget(self.warning)  self.gridLayout.addLayout(self.verticalLayout\_4, 0, 0, 1, 1)  weather.setCentralWidget(self.centralwidget)  self.statusbar = QtWidgets.QStatusBar(weather)  self.statusbar.setObjectName("statusbar")  weather.setStatusBar(self.statusbar)  self.retranslateUi(weather)  self.select.clicked['QModelIndex'].connect(weather.get\_weather)  self.pushButton.clicked.connect(weather.get\_weather\_list)  QtCore.QMetaObject.connectSlotsByName(weather)  def retranslateUi(self, weather):  \_translate = QtCore.QCoreApplication.translate  weather.setWindowTitle(\_translate("weather", "MainWindow"))  self.pushButton.setText(\_translate("weather", "search"))  main.py  # coding:utf-8  import sys  import weather\_com\_cn as w  import weather  from PyQt5.QtWidgets import QMainWindow, QApplication  from PyQt5.QtGui import QPixmap  class MainWindow(QMainWindow):  city\_dict = []  city\_code = -1  def \_\_init\_\_(self, parent = None):  super(QMainWindow, self).\_\_init\_\_(parent)  self.ui = weather.Ui\_weather()  self.ui.setupUi(self)    def get\_weather\_list(self):  try:  self.ui.warning.setText("please wait...")  city = w.search(self.ui.search.toPlainText())  self.ui.select.clear()  self.city\_dict = []  for k,v in city.items():  self.city\_dict.append(v)  self.ui.select.addItem(k)  self.ui.warning.setText("done!")  except:  self.ui.warning.setText("an error happens while fetching list!")    def get\_weather(self):  self.ui.warning.setText("please wait...")  try:  #print(self.ui.select.currentRow())  self.city\_code = self.city\_dict[self.ui.select.currentRow()]  #print(self.city\_code)  info = w.get\_weather(self.city\_code)  #print(info)  self.ui.time0.setText(info[0]['time'])  self.ui.wea0.setText(info[0]['wea'])  pix = QPixmap(r'./pic/' + info[0]['weap'] + r'.png')  self.ui.weap0.setPixmap(pix)  self.ui.tem0.setText(info[0]['tem'])  self.ui.win0.setText(info[0]['win'])  pix = QPixmap(r'./pic/' + info[0]['winp'] + r'.png')  self.ui.winp0.setPixmap(pix)  self.ui.suntime0.setText(info[0]['suntime'])  # night  self.ui.time1.setText(info[1]['time'])  self.ui.wea1.setText(info[1]['wea'])  pix = QPixmap(r'./pic/' + info[1]['weap'] + r'.png')  self.ui.weap1.setPixmap(pix)  self.ui.tem1.setText(info[1]['tem'])  self.ui.win1.setText(info[1]['win'])  pix = QPixmap(r'./pic/' + info[1]['winp'] + r'.png')  self.ui.winp1.setPixmap(pix)  self.ui.suntime1.setText(info[1]['suntime'])  self.ui.warning.setText("done!")  except:  self.ui.time0.clear()  self.ui.wea0.clear()  self.ui.weap0.clear()  self.ui.tem0.clear()  self.ui.win0.clear()  self.ui.winp0.clear()  self.ui.suntime0.clear()  self.ui.time1.clear()  self.ui.wea1.clear()  self.ui.weap1.clear()  self.ui.tem1.clear()  self.ui.win1.clear()  self.ui.winp1.clear()  self.ui.suntime1.clear()    self.ui.warning.setText("404 not found")    if \_\_name\_\_ == '\_\_main\_\_':  myapp = QApplication(sys.argv)  mywin = MainWindow()  mywin.show()  sys.exit(myapp.exec\_())  **实验结果：（贴运行截图）**  **爬取的网页：**  <http://www.weather.com.cn/weather1d/101070102.shtml#input>    程序运行截图：    没有互联网连接时：    当爬取的网页无响应时：    无响应原因：[www.weather.com.cn](http://www.weather.com.cn) 对于小型乡镇等跳转的网页与中大城市使用的网页不同。  资源文件来源：  通过爬取网页得到原始图片，根据网页渲染原理进行图片分割保存备用。  blue80  分割后图片：    **实验题目 第二个：**  **制作一个简单的舆情热词分析程序：输入监测的新闻网站列表，通过分析导出当前最热门的新闻词汇，并用图文并茂的形式展示出来。**  **对应源代码：**  通过re库爬取输入监测的网站的所有a标签并提取其中的文字，然后通过统计程序和jieba 分词分析以及wordcloud 库生成一个词云，并显示在界面上，从而一目了然、图文并茂显示最热门的新闻词汇。  热词分析模块在filter.py 文件中实现，界面同样使用PyQt5 和 pyqt5\_tools 实现。  filter.py  # coding:utf-8  def get\_labels(url):  '''  input: the link of the web and the decode of the website(UTF-8 default)  output: the list of the sentences  '''  import requests  import jieba  import re  from bs4 import BeautifulSoup  headers = {  'User-Agent':'Mozilla/5.0 (Windows NT 10.0; Win64; x64) '  'AppleWebKit/537.36 (KHTML, like Gecko) Chrome/70.0.3538.102 Safari/537.36'  }  rq = requests.get(url, headers = headers)  html = rq.content  bs\_obj = BeautifulSoup(html, "html.parser")  labels = bs\_obj.find\_all('a')  except\_dict = ['\+', '\?', '\!', '1', '2', '3', '4', '5', '6', '7', '8', '9', '0', '\n', '\t']  useless\_list = [r'凤凰', r'新浪', r'新闻', r'独家', r'资讯', r'凰家尚品', r'体育文化', r'财经', r'图', r'娱乐', r'视频', r'体育', r'小说', r'视频', r'时尚', r'汽车房产']  results = []  for j in labels:  text = ""  for i in except\_dict:  # print(j.text)  text = re.sub(i, '', j.text)  for i in useless\_list:  text = text.replace(i, "")  results.append(text)  return results  def count\_label(results):  '''  input: the list of the sentences  output: a dictionary of the frequency of the worlds  '''  import jieba  words = []  for i in results:  tmp = jieba.lcut(i)  words = words + tmp  summary = {}  for i in words:  if not i in summary.keys():  summary[i] = 1  else:  summary[i] += 1  return summary  def connect\_sentences(short\_sentences\_dir):  text = ""  for i in short\_sentences\_dir:  text = text + i + " "  return text  def generate\_word\_cloud(input\_sentence, width = 1000, height = 800, background\_color = 'white', font\_path = "simkai.ttf", file\_path = 'word\_cloud.png'):  '''  input: the long text (seperated by space between words)  output: the picture of the words  '''  import wordcloud  w = wordcloud.WordCloud(width=width,height=height,background\_color=background\_color, font\_path = font\_path)  w.generate(input\_sentence)  w.to\_file(file\_path)  analyze.py  # -\*- coding: utf-8 -\*-  # Form implementation generated from reading ui file 'analyze.ui'  #  # Created by: PyQt5 UI code generator 5.15.2  #  # WARNING: Any manual changes made to this file will be lost when pyuic5 is  # run again. Do not edit this file unless you know what you are doing.  from PyQt5 import QtCore, QtGui, QtWidgets  class Ui\_analyze(object):  def setupUi(self, analyze):  analyze.setObjectName("analyze")  analyze.resize(800, 600)  analyze.setMinimumSize(QtCore.QSize(800, 600))  self.centralwidget = QtWidgets.QWidget(analyze)  self.centralwidget.setObjectName("centralwidget")  self.gridLayout = QtWidgets.QGridLayout(self.centralwidget)  self.gridLayout.setObjectName("gridLayout")  self.horizontalLayout = QtWidgets.QHBoxLayout()  self.horizontalLayout.setObjectName("horizontalLayout")  self.verticalLayout = QtWidgets.QVBoxLayout()  self.verticalLayout.setObjectName("verticalLayout")  self.input = QtWidgets.QTextEdit(self.centralwidget)  self.input.setMinimumSize(QtCore.QSize(260, 400))  self.input.setMaximumSize(QtCore.QSize(260, 400))  self.input.setObjectName("input")  self.verticalLayout.addWidget(self.input)  self.generate = QtWidgets.QPushButton(self.centralwidget)  self.generate.setMinimumSize(QtCore.QSize(260, 100))  self.generate.setMaximumSize(QtCore.QSize(260, 100))  self.generate.setObjectName("generate")  self.verticalLayout.addWidget(self.generate)  self.horizontalLayout.addLayout(self.verticalLayout)  self.verticalLayout\_2 = QtWidgets.QVBoxLayout()  self.verticalLayout\_2.setObjectName("verticalLayout\_2")  self.wordcloud = QtWidgets.QLabel(self.centralwidget)  self.wordcloud.setMinimumSize(QtCore.QSize(400, 400))  self.wordcloud.setMaximumSize(QtCore.QSize(400, 400))  self.wordcloud.setAutoFillBackground(True)  self.wordcloud.setText("")  self.wordcloud.setObjectName("wordcloud")  self.verticalLayout\_2.addWidget(self.wordcloud)  self.count = QtWidgets.QListWidget(self.centralwidget)  self.count.setMinimumSize(QtCore.QSize(400, 100))  self.count.setMaximumSize(QtCore.QSize(400, 100))  self.count.setObjectName("count")  self.verticalLayout\_2.addWidget(self.count)  self.horizontalLayout.addLayout(self.verticalLayout\_2)  self.gridLayout.addLayout(self.horizontalLayout, 0, 0, 1, 1)  analyze.setCentralWidget(self.centralwidget)  self.statusbar = QtWidgets.QStatusBar(analyze)  self.statusbar.setObjectName("statusbar")  analyze.setStatusBar(self.statusbar)  self.retranslateUi(analyze)  self.generate.clicked.connect(analyze.generate)  QtCore.QMetaObject.connectSlotsByName(analyze)  def retranslateUi(self, analyze):  \_translate = QtCore.QCoreApplication.translate  analyze.setWindowTitle(\_translate("analyze", "MainWindow"))  self.generate.setText(\_translate("analyze", "GENERATE"))  main.py  # coding:utf-8  import sys  import filter as f  import analyze  from PyQt5.QtWidgets import QMainWindow, QApplication  from PyQt5.QtGui import QPixmap  class MainWindow(QMainWindow):  def \_\_init\_\_(self, parent = None):  super(QMainWindow, self).\_\_init\_\_(parent)  self.ui = analyze.Ui\_analyze()  self.ui.setupUi(self)    def generate(self):  try:  sentences = []  summary = {}  #print(self.ui.input.toPlainText())  self.ui.wordcloud.clear()  self.ui.count.clear()  input\_url = self.ui.input.toPlainText()  input\_url = input\_url.split('\n')  for i in input\_url:  sentences = sentences + f.get\_labels(i)  summary = f.count\_label(sentences)  for k, v in summary.items():  self.ui.count.addItem(str(k) + ": " + str(v))  tmp = f.connect\_sentences(sentences)  f.generate\_word\_cloud(tmp, width = 400, height = 400)  pix = QPixmap(r'./word\_cloud.png')  self.ui.wordcloud.setPixmap(pix)  except:  import traceback  traceback.print\_exc()  if \_\_name\_\_ == '\_\_main\_\_':  myapp = QApplication(sys.argv)  mywin = MainWindow()  mywin.show()  sys.exit(myapp.exec\_())  **实验结果：（贴运行截图）**    word_cloud  通过输入网页并爬取分析关键词，生成词云图并展示在界面中。 | | | | |