**实验2：文件信息采集器实验报告**

姓名：江继军 学号：21373222 班级：213921

实验日期：2024年 3 月 20 日 实验地点：一号馆410 成绩：

1. **实验目的**
2. 基于Python-docx，编写一个文件信息采集器，将选定的文本内容，例如实验一爬取的内容，从其他类型文件写入Word文件，加入对关键词的搜索、统计或标注。
3. 通过此实验，初步熟悉常见文件信息采集的Python编程。

**二、实验内容**

1. 环境安装

1. 安装Python编程环境，即VScode或pycharm-Professional/community, 以及Python中的相关库。

2文件内容采集器程序和文件内容提取

1. 编制文件信息采集器程序，此程序可以选择输入的文件路径和类型。
2. 对内容进行识别提取，以新闻类数据为例。导入内容可包含但不限于：文章标题（Title）、发表日期与时间表（Date time）、正文（Content）。各部分内容均另起一段，正文按自然段存储，文件易读且没有乱码。。
3. 对文件进行解析，根据自定义的关键词，给出统计或标记。

**三、实验环境**

分项列出主要实验环境（主机/操作系统、Python开发环境、Python版本、Scrapy库版本、其他库版本等，举例如下）：

1. 操作系统：Win11
2. Python开发环境：pycharm-community
3. Python版本：3.9
4. Python-docx库版本：1.1.0
5. Matplotlib版本：3.5.2
6. reportlab版本：4.1.0

**四、实验步骤**

1. 文件采集器编程实例（写明编写的代码或使用的命令，需有注释）。加分项：额外生成PDF文件，且格式正确无乱码，+3分；统计后做可视化展示，+2分。
2. 收集文件信息

首先根据要求目录路径和文件类型收集文件，使用os.walk()函数将文件相对路径保存



图 1 收集文件信息函数

1. 识别和提取文件内容

根据所需文件的特殊格式，识别并将里面的内容保存为Word文件

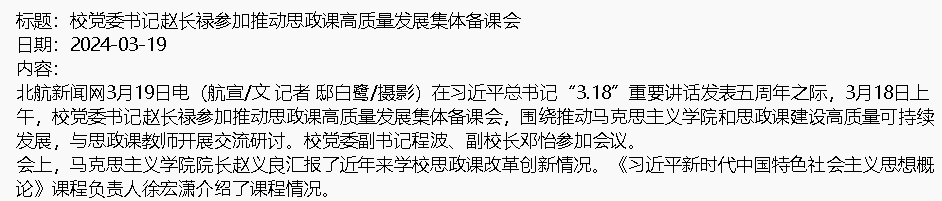


图 2 新闻文件内容格式





图 3 识别与提取文件内容

1. 解析文件内容

根据关键词字典，将新闻里的关键词高亮，并统计结果，docx文件见下第2问



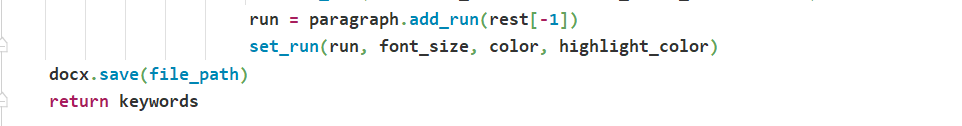


图 4 解析文件内容

1. 将Word文档转换成PDF文档

用到reportlab库进行转换，注意中文乱码问题，需将字体文件放在Python安装路径下的/site-packages/reportlab/fonts文件夹中

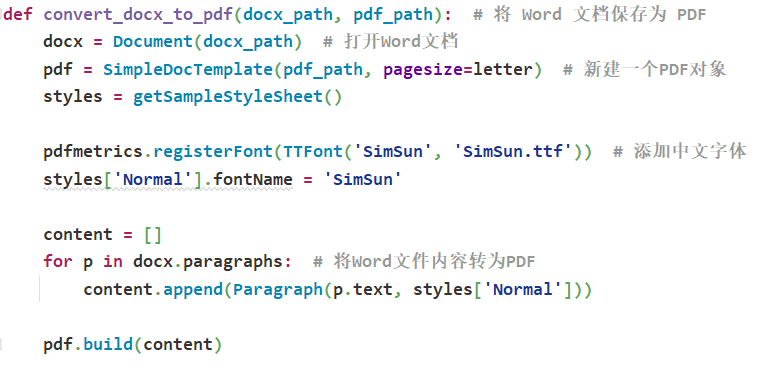


图 5 Word转PDF

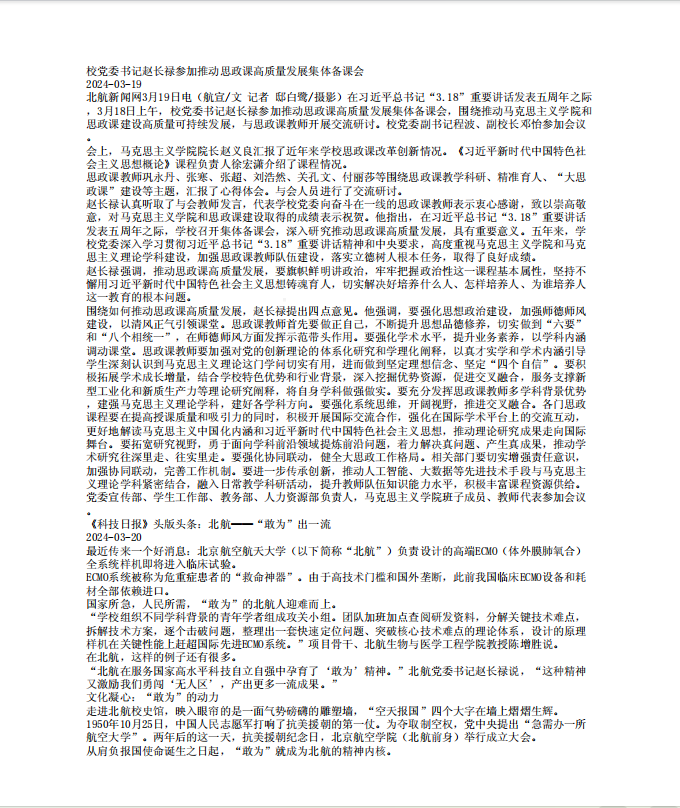


图 6 PDF内容

1. 统计结果展示

使用matplotlib库将结果可视化展示，见下第3问，并将结果记录在result.txt文件中



图 7 结果可视化展示



图 8 结果保存文件

1. DOCX源文件截图（可缩小，但需清晰）。

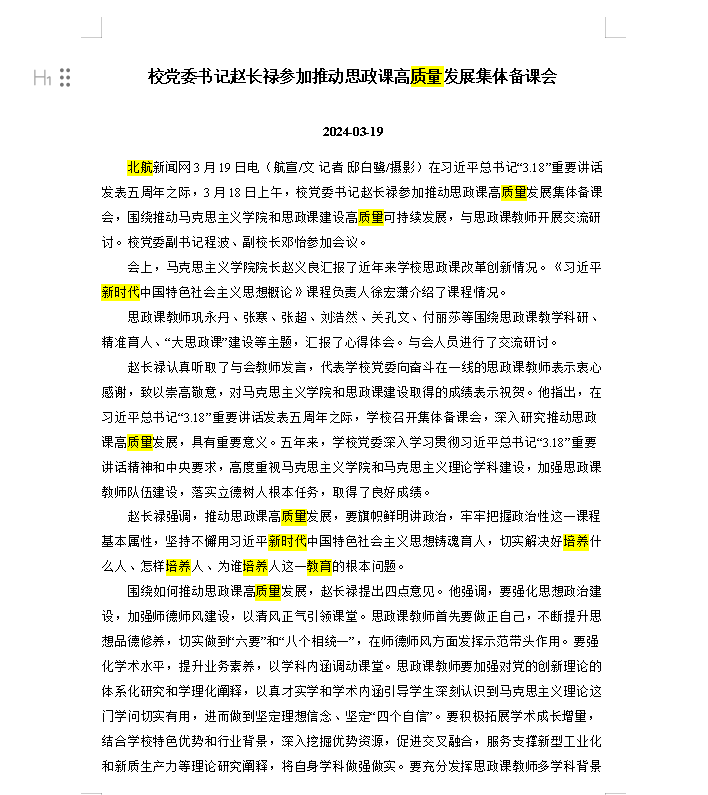


图 9 Word源文件内容

1. 给出最后关键信息统计或标注结果文件的截图（可缩小，但需清晰）

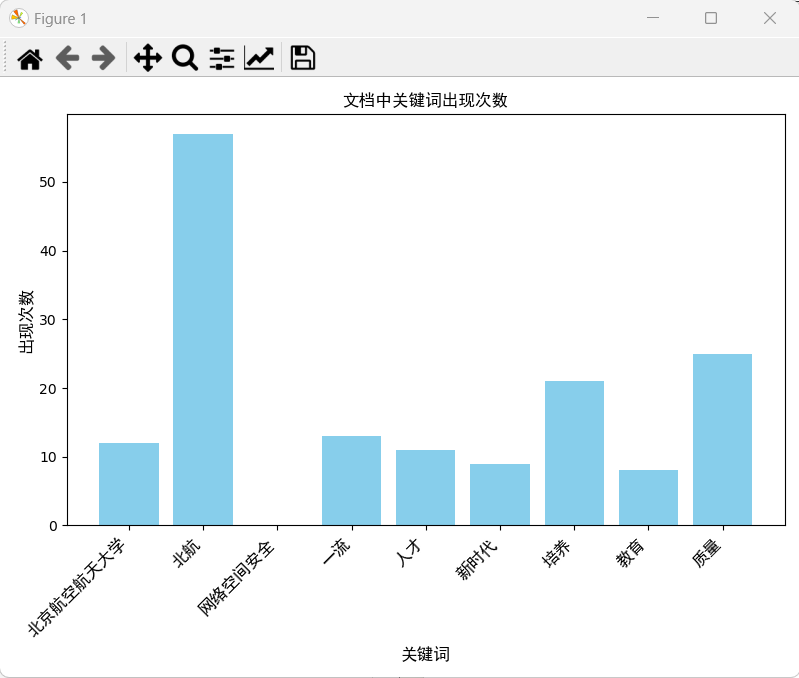


图 10 结果可视化展示

**五、问题和解决方法**

1. Word中文字体问题，在使用python-docx库设置中文字体时，需要先设置英文字体run.font.name = u'Times New Roman'

然后再设置中文字体

run.\_element.rPr.rFonts.set(qn('w:eastAsia'), u'宋体')

2. 在Word转PDF中，也要注意字体问题，需要将字体文件保存到对应的库中

3. 使用PyMuPDF库生成PDF文件困难，网上搜索资料使用其他的库生成PDF文件。

**六、实验总结**

通过本实验，我学会了如何使用Python-docx库来处理Word文档，并且了解了如何使用os库编写一个简单的文件信息采集器。同时，我也掌握了如何进行文本内容的识别提取和关键词的统计或标记，这对于处理大量文本数据具有很大的帮助。在未来的工作中，我将会进一步完善这个文件信息采集器，使其能够处理更多类型的文档，并且增加更多功能，提升其实用性和可扩展性。

**七、附件**

提交可执行的原程序压缩包，随电子版实验报告提交，文件名称为学号+姓名+实验二。