**根据关键词保留语句**

继续分享上次在比赛的过程中的一个小问题的解决方案，以后大家遇到类似的问题就可以直接使用了。

在开始之前，打一个GuangGao，我的某鱼上上新了一些技术图书，图片中的书籍都是正版的书籍，没有影印的，有一些是全新的未拆封的，有兴趣的小伙伴可以来围观一下，想要购买的可以在某鱼上跟我私聊获取价格，多买多优惠。

**闲鱼账号:客串太潦草**



接下开就开始我们今天的内容

1. **需求分析**

最近在参加人工智能建模大赛的时候，需要对一篇文章中的语句进行清洗，清洗的规则就是：基于一个关键词表来对文本语句进行筛选，使用逗号替换句号、分号或者是顿号。最终保留一些我们需要的语句，也就是命中关键词的语句。

具体来说就是，现在假设我们的文本如下



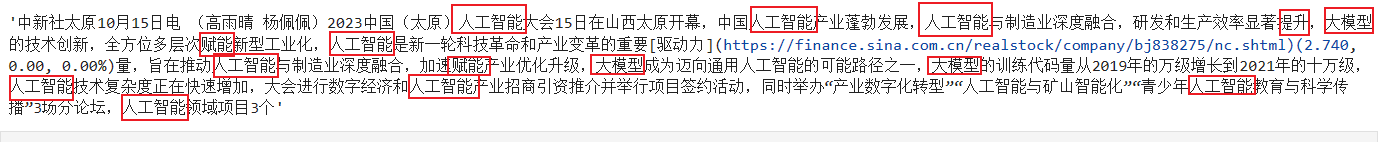
注意的是，各行文本之间是分段的，我们使用Typore打开该文件，其样式如下：



我们的关键词表语句如下，我们这里直接使用DataFrame进行存储，大家在实际的项目中可以存在诸如CSV文件中：



最后我们的需求就是清洗之后得到以下的文档内容：



1. **实现**

for 循环实现这个功能没有什么技术含量，大家可以试试，这里就不在进行阐述了。我们主要讲解怎么使用python中的any()函数来实现这个功能。

首先我们简单的聊聊any()函数，在Python中，any()函数是一个内置函数，用于判断可迭代对象中是否存在**任何一个**元素为真（非零、非空、非None）的情况。

该方法的语法为any(iterable)，其中，iterable是一个可迭代对象，比如说我们常见的列表、元组、集合、字典的键等。

接下里我们举几个例子，看看这个any()方法：

numbers = [0, 1, 2, 3, 4]

result = any(numbers)

print(result)

# 输出 True

因为列表numbers中存在不为0的元素

numbers = [0, 1, 2, 3, 4]

result = any(num>4 for num in numbers )

print(result)

# 输出 False

因为列表numbers中不存在大于4的元素

实际的例子还是很简单的，相对应的我们还有一个all()函数，all()函数与any()函数有着很大类似之处，用于判断可迭代对象中是否存在**所有的**元素为真（非零、非空、非None）的情况。

接下来我们开始进行实现，代码如下：

import pandas as pd

kw\_df = pd.DataFrame({'id': [1, 2, 3, 4], 'kw': ['人工智能','提升','赋能','大模型']})

# 将关键词表转换为列表的形式

kw\_list = list(kw\_df['kw'])

# 加载文本,并去除符号

with open('清洗文本.txt', 'r',encoding='UTF8') as file:

# 读取文件内容并去除换行符

text = file.read().replace('\n', '')

# 符号替换,将其他的符号转换为逗号,这样来拆分语句

text = text.replace('。','，')

text = text.replace('、','，')

# 对文本进行处理

# 先使用逗号进行分割文本，然后使用any()函数进行判断

def keep\_sentence(text,keep\_list):

result = [item for item in text if any(each in item for each in keep\_list)]

return '，'.join(result)

res = keep\_sentence(text.split('，'),kw\_list)

res



这样我们就完成了，文本的提取功能。

如果我们的需求是，获取不包含这些关键的语句，则我们的代码只需要修改其中一处即可，即将result那张修改为：

result = [item for item in text if not any(each in item for each in keep\_list)]

即加上not关键词即可，大家可以自己试一试。

1. **总结**

以上就是今天文章的全部内容了，下次如果有小伙伴有类似的需求可以采取上述方案。另外，大家可以想想有什么其他的方案来实现这个需求。我们下次再见。