**文本分析**

[一、 文本分析的定义 2](#_Toc27383265)

[二、 文本分析的基础理论知识 2](#_Toc27383266)

[**1.** **语义分析与语法分析** 2](#_Toc27383267)

[**2.** **文本的结构化分析** 2](#_Toc27383268)

[**3.** **文本的标准化分析** 3](#_Toc27383269)

[三、 文本分析的主要术语及其原理 3](#_Toc27383270)

[四、 分本分析的主要应用 8](#_Toc27383271)

[五、 以红楼梦为例的简单应用 8](#_Toc27383272)

[**1.** **数据与分析工具** 8](#_Toc27383273)

[**2.** **分词** 8](#_Toc27383274)

[**3.** **词性标注** 9](#_Toc27383275)

[**4.** **基于TF-IDF算法的关键词提取** 10](#_Toc27383276)

1. 文本分析的定义

文本分析是指对文本的表示及其特征项的选取；文本分析是文本挖掘、信息检索的一个基本问题，它把从文本中抽取出的特征词进行量化来表示文本信息。文本（text），与讯息（message）的意义大致相同，指的是由一定的符号或符码组成的信息结构体，这种结构体可采用不同的表现形态，如语言的、文字的、影像的等等。文本是由特定的人制作的，文本的语义不可避免地会反映人的特定立场、观点、价值和利益。因此，由文本内容分析，可以推断文本提供者的意图和目的。

1. 文本分析的基础理论知识
2. **语义分析与语法分析**

从技术层面来看，文本分析任务主要分为语义分析和语法分析。有些文本分析任务只有其一，有些文本分析任务则兼具二者的特征。语义分析和语法分析具有截然不同的研究框架。语义分析的主要目的是弄清楚段文本内容大体上讲述了什么主题；语法分析的主要目的是要弄清楚文本中各个语言要素之间的组成结构关系。

语义分析可以有不同的应用层次，即在分析粒度上的差异。对于一段文本内容，语义分析既可以很细致，也可以很粗糙。如果进行粗糙的语义分析只需要与语义分析相关的技术即可，而如果进行细致的语义分析，就要配合使用用语法分析的技术手段。如果进行粗糙的语义分析只需要与语义分析相关的技术即可,而如果进行细致的语义分析,就要配合使用用语法分析的技术手段。

语义分析的典型应用情境有如下几种：

1. 判断两个词汇(或两篇文档)的语义相似性。
2. 将文本要素(词汇,句子、文章)进行结构转化。
3. 提取文章中的关键文本信息。
4. 提取文章中的主题信息。

......

语法分析的典型应用情境有如下几种：

1. 判断句子中词汇的词性。
2. 判断句子的语法结构,构建语法树。
3. 判断句子中代表特定角色的词汇或短语。

④基于语料库自动构建知识网络(本体或关系数据库)。

......

1. **文本的结构化分析**

文本类型数据是非结构化的数据，无论是管理类应用还是内容类应用，都只能对结构化的数据进行处理。因此，在文本挖掘中非常重要的技术环节就是数据的结构化过程。基于结构化的数据形式，可以进行统计建模分析，并在计算机中进行自动化处理。文本分析任务需要对文本中基本要素进行结构化处理，因此，需要先对文本数据中的基本要要素进行定义：

图2.2.1 文本分析基本要素框架

* 1. 字（字符）：字（字符）是文本内容的最小组成单元。中文的文本分析任务通常是字，英文的文本分析任务通常是字符。字（字符）通常难以独立构成语义，因此大多数文本分析任务不以字为最基本的分析单元。
  2. 词汇（词组）：词汇是由字组成的，是具有特定语言含义的最小单元，因此大多数文本分析任务将词汇作为最基本的分析单元。
  3. 句子（短文本）：句子是由词或词组进一步组成的，具有一定的语法结构，通常在对文本进行语法分析时需要将句子作为主要的研究对象。
  4. 文档：文档是由句子组成的，对文本进行分析大多是针对文档进行分析。
  5. 语料库：语料库是由很多文档组成的。对语料库进行分析，有利于更好地了解词汇、句子及文章的内容含义。

1. **文本的标准化分析**

与文本挖掘技术关系十分密切的另外一个概念就是文本标准化。语言是非常灵活的一种信息形式,这导致其表现形式通常是杂乱无章的,这为文本内容分析带来诸多不便与困扰。因此，在进行文本分析之前,一般需要对其进行标准化处理，增加文本类型数据的一致性、整洁性。对文本进行标准化处理，可以降低后续文本建模、挖掘、分析等任务的难度，并提升结果的准确度。文本的标准化相当于数据挖掘技术的数据预处理阶段，一般在文本结构化任务之前。对文本进行标准化处理的具体方法很多，其依赖于所分析的文本集合的特征。很多标准化方法是基于丰富的数据处理经验进行设计与构建的,当前常用的文本信息标准化方法有以下几种：

1. 对文本集合的语言进行统一，去除其他语种文章。
2. 去除无效符号、特殊符号，以及网页标记。
3. 简体字与繁体字的统一(中文文本)。
4. 去除文本显示格式信息(如空格、回行等)。
5. 对文本中的错别字自动修复。
6. 构建同义词或近义词表，减少文本特征。
7. 过滤低频(词汇出现次数很低)的文本特征。
8. 文本分析的主要术语及其原理

1.节点（顶点）：在文本分析中，一个节点代表一个人物或事物。

2.度：节点的度定义为与该节点相连的边的数目。

3.词云：由词汇组成类似云的彩色图形。“词云”就是对网络文本中出现频率较高的“关键词”予以视觉上的突出，形成“关键词云层”或“关键词渲染”，从而过滤掉大量的文本信息，使浏览网页者只要一眼扫过文本就可以领略文本的主旨。

4.气泡线（Bubblelines）：语料库中的每个文档均以一条水平线表示，并分为等长的段（默认为50段）。每个选定的单词都表示为气泡，气泡的大小指示相应文本段中单词的频率。气泡越大，单词出现的频率就越高。

5.气泡（Bubbles）：气泡是按文档形式对字词频率的生动呈现。高频字词按文档顺序读取，当前字词显示在右上角。下面的字词列表显示了文档中该点的累积频率等级。第一次遇到字词时，会创建一个气泡并将其添加到画布的主要部分。该字词在读取时以黄色闪烁。气泡的相对大小表示相对于文档中当前点的其他字词而言的频率。

6.语料库：语料库语料库是由很多文档组成的。对语料库进行分析，有利于更好地了解词汇、句子及文章的内容含义。对于给定的文本内容，虽然人可以对其内涵进行有效的判断，但是计算机只是将其作为般的符号来处理。计算机本身不含有对文本进行理解的先验知识，因此只能通过“阅读”大量的文档来学习和文本有关的知识。

语料库本质上就是计算机学习的文本资料，让计算机通过“阅读”大量语料库里的文章来学习如何去从语义层面和语法层面对特定层次的文本单元进行理解。广义的语料库除了包括文档的集合，还包含任何计算机都可以进行统计学习的知识集合。

语料库通常分为通用语料库和专用语料库。通用语料库与文本分析的具体应用场景无关，通常仅按照语言进行划分，如英语的通用语料库、汉语的通用语料库等。通用语料库通常大而全，只要是某种语言的文本，无论什么内容都会有所涉及。通过对通用语料库进行分析，可以了解某种语言最基础的特征，相应的结论可以用于解决一般的、 没有特殊需求的文本分析任务。

专用语料库通常与某种具体的应用场景的相关性较高，与某种特定的文本分析任务密切相关的语科库都被称为专用语料库，其包含的内容五花八门且针对性强，有口语语料库、书面语语料库、方言语料库、在线评论语料库、新闻语料库、医疗语料库等。

7.词频（TF）：在一份给定的文件里，**词频**（term frequency，TF）指的是某一个给定的词语在该文件中出现的次数。这个数字通常会被正规化，以防止它偏向长的文件(同一个词语在长文件里可能会比短文件有更高的词频，而不管该词语重要与否。)

对于在某一特定文件里的词语来说，它的重要性可表示为：

  以上式子中是该词在文件中的出现次数，而分母则是在文件中所有字词的出现次数之和。

8.逆向文件频率（IDF）：逆向文件频率（inverse document frequency，IDF）是一个词语普遍重要性的[度量](https://baike.baidu.com/item/%E5%BA%A6%E9%87%8F" \t "_blank)。某一特定词语的IDF，可以由总文件数目除以包含该词语之文件的数目，再将得到的商取对数得到。

其中，|D|：语料库中的文件总数；：包含词语的文件数目（即的文件数目）如果该词语不在语料库中，就会导致被除数为零，因此一般情况下使用 ；然后

9.词频-反转文件频率（term frequency–inverse document frequency，TF-IDF）：词频--反转文件频率，是一种用于情报检索与文本挖掘的常用加权技术，用以评估一个词对于一个文件或者一个语料库中的一个领域文件集的重要程度。字词的重要性随着它在文件中出现的次数成正比增加，但同时会随着它在语料库中出现的频率成反比下降。TF-IDF加权的各种形式常被搜索引擎应用，作为文件与用户查询之间相关程度的度量或评级。

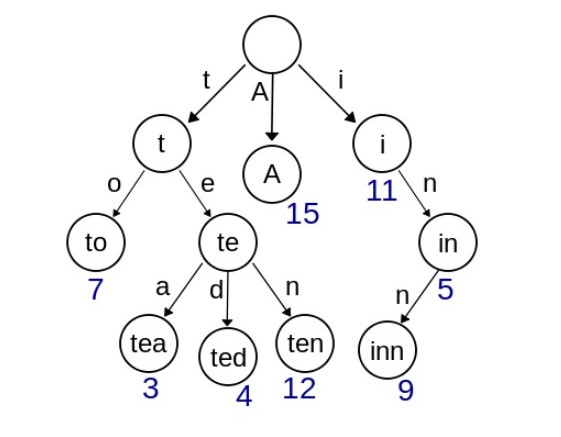
某一特定文件内的高词语频率（TF），以及该词语在整个文件集合中的低文件频率(IDF)，可以产生出高权重的TF-IDF。因此，TF-IDF倾向于过滤掉常见的词语，保留重要的词语。

**TF-IDF的主要思想是：**如果某个词或短语在一篇文章中出现的频率TF高，并且在其他文章中很少出现，则认为此词或者短语具有很好的类别区分能力，适合用来分类。TFIDF实际上是：TF \* IDF，TF词频(Term Frequency)，IDF反文档频率(Inverse Document Frequency)。**TF**表示词条在文档d中出现的频率（另一说：**TF词频(Term Frequency)指**的是某一个给定的词语在该文件中出现的次数）。

定义说明：如果包含词条t的文档越少，也就是n越小，IDF越大，则说明词条t具有很好的类别区分能力。如果某一类文档C中包含词条t的文档数为m，而其它类包含t的文档总数为k，显然所有包含t的文档数n=m+k，当m大的时候，n也大，按照IDF公式得到的IDF的值会小，就说明该词条t类别区分能力不强。（另一说：**IDF反文档频率(Inverse Document Frequency)**是指如果包含词条的文档越少，IDF越大，则说明词条具有很好的类别区分能力。）

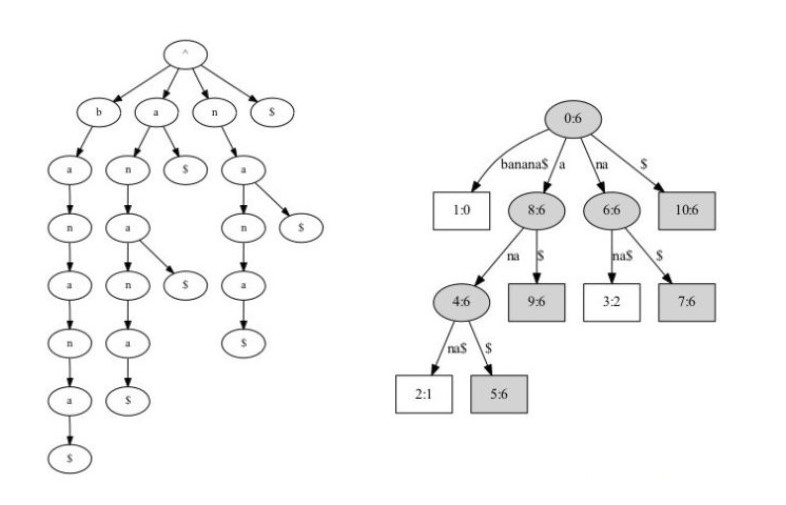
缺点：如果一个词条在一个类的文档中频繁出现，则说明该词条能够很好代表这个类的文本的特征，这样的词条应该给它们赋予较高的权重，并选来作为该类文本的特征词以区别与其它类文档，但是按照TF-IDF计算会得到较低的权重，这就是IDF的不足之处.

10. 字典树：图中，每个圆圈是一个结点，代表着一个字符串（就是圆圈内的内容）；结点之间的连线是边，代表着一个字母。最上面的结点，也就是空着的那个结点，是根结点。如果我们从根结点不断向下走到某个结点，那么把经过的每一条边上的字母拼起来，就是这个结点代表的字符串了。这就是字典树的特点。那么字典树是干什么用的呢？举个例子来说，假如我们想在这棵字典树里查找 “to” 这个单词，就可以先从根结点下面的边里找到第一个字母，也就是 “t” 这条边，从而找到 “t” 这个结点。然后我们再从 “t” 结点下面的边里找到第二个字母，也就是 “o” 这条边，就找到 “to” 这个结点了。假如 “to” 这个结点里储存了 “to” 的中文解释，那么我们只通过两次操作就找到了 to 的中文意思。这样比一个词一个词地找的方法快多了。这很像我们查字典的时候，先看第一个字母在字典中的位置，然后再看第二个字母……最终找到单词，因此被称为字典树。



11. 后缀树和后缀字典树

后缀字典树，其实就是字典树，只不过里面的内容不是单词，而是一个字符串的所有后缀：从第一个字母到最后一个字母的内容，从第二个字母到最后一个字母的内容……以此类推。比如说，"banana" 的所有后缀就是 banana, anana, nana, ana, na 和 a。把这些内容都加到字典树里，就构成了后缀字典树。下面左图就是 banana 的后缀字典树：



后缀树和后缀字典树的区别就是，在后缀树中，我们要把下面只有一条边的结点去掉，然后把这个结点连接的两条边压缩成一条。比如，左图后缀字典树中的 b-a-n-a-n-a，在右图的后缀树中被压缩成了 banana 这一条边。此外，后缀树还使用了一个技巧，就是不储存边的内容，而是储存这些内容在原文中的位置。因为后缀树中的很多内容都是重复的，所以这个小技巧可以大大减少索引的大小（用专业的语言描述，它的空间复杂度是 O(n)）。

后缀树又有什么用呢？它最大的用途就是检索字符串中间的内容。比如，假如我想查找 an 在 banana 中哪里出现过，只需要查找代表 an 的结点，就找到了所有以 an 开头的结点： anana 和 ana。由于每次出现 an 的地方都一定会产生一个以 an 开头的后缀，而所有的后缀都在后缀树中，所以这样一定能够找到所有 an 出现的位置。后缀树的强大之处在于，即使我们把 banana 换成一篇很长很长的文章，我们也能很快地进行这样的检索。

12. 凝固度：一个片段出现的频率比左右两部分分别出现的频率的乘积高出多少倍（注意，频率表示的是出现的比例，而频数表示的是出现的次数）。

公式：如果 P(AB) 是片段出现的频率，P(A) 是片段左边的字的出现的频率， P(B) 是右边的字出现的频率，那么凝固度 co 就是：

公式中，P(A)\*P(B) 就是左右部分在完全随机组合的情况下被组合到一起的概率。凝固度的思想是：如果片段实际出现的概率比被随机组合出来的概率高出很多倍，就说明这样的组合应该不是意外产生的，而是有一些关联的。这个关联很可能就是因为这个片段是一个不可分割的整体，也就是单词。对于超过两个字的片段，可以尝试每一种拆分方法（比如“贾宝玉”有“贾/宝玉”和“贾宝/玉”两种拆分方法），然后取各种方法的凝固度的最小值。

13. 自由度：为了排除掉不完整的单词，我们可以使用**自由度**这个概念来继续过滤。自由度的思想是这样的：如果一个组合是一个不完整的单词，那么它总是作为完整单词的一部分出现，所以相邻的字就会比较固定。比如说，“香院”在原文中出现了 23 次，而“梨香院”出现了 22 次，也就是说“梨”在“香院”的左边一起出现的频率高达 95.7%，所以我们有把握认为“香院”不是完整的单词。而自由度描述的就是一个片段的相邻字有多么的多样、不固定。如果片段的自由度比较高，就说明这个词应该是完整的反之，就是不完整的。

14. 停用词（Stopwords）: 停用词是指在信息检索中，为节省存储空间和提高搜索效率，在处理自然语言数据（或文本）之前或之后会自动过滤掉某些字或词，这些字或词即被称为Stop Words（停用词）。这些停用词都是人工输入、非自动化生成的，生成后的停用词会形成一个停用词表。但是，并没有一个明确的停用词表能够适用于所有的工具。甚至有一些工具是明确地避免使用停用词来支持短语搜索的。

对于一个给定的目的，任何一类的词语都可以被选作停用词。通常意义上，停用词大致分为两类。一类是人类语言中包含的功能词，这些功能词极其普遍，与其他词相比，功能词没有什么实际含义，比如'the'、'is'、'at'、'which'、'on'等。称它们为停用词是因为在文本处理过程中如果遇到它们，则立即停止处理，将其扔掉。停用词主要包括英文字符、数字、数学字符、标点符号及使用频率特高的单汉字等。

目前常用的停用词表：哈工大停用词表、四川大学机器智能实验室停用词库、百度停用词表等。（停用词表附载链接：<https://github.com/goto456/stopwords>）

15.命名实体：是指人名、地名、机构名等。人名如张三、李四、王五；地名如青岛、济南、青岛市黄岛区辛安街道；结构名如清华大学，山东大学，山东科技大学。命名实体可看作一个词，若其搭配无法在词法分析系统构建则全部收集，应用命名实体识别技术帮助识别。其他命名实体还包括商品名、武器名等。

1. 分本分析的主要应用

中文分词、词性标注与序列标注、命名实体识别、文本分类、文本聚类、文本检索、垃圾邮件过滤与情感分析、社交网络分析、口碑分析和敏感词识别等。

1. 以红楼梦为例的简单应用
2. **数据与分析工具**

1.1 数据来源

①人民文学出版社出版的《红楼梦》庚辰本的部分语句，以这些语句为例子进行简单的应用分析。

②网络下载的《红楼梦.txt》。

1.2分析工具

扩展jieba中文分词模块的Python3.6.6。（jieba中文分词模块网址：<https://github.com/fxsjy/jieba>）

1.3语料库

1.3.1语料库在线提供在线查询并下载语料库，同时还提供了相关的资源和工具，也可以在线分析语句。（网址：<http://corpus.zhonghuayuwen.org/index.aspx>）

1.3.2BCC语料库提供在线查询并下载语料库，资源更加丰富，而且支持多个领域，还支持自定义筛选等多种功能（网址：<http://bcc.blcu.edu.cn/>）

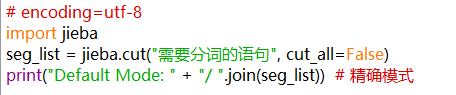
1. **分词**

2.1分词概念

分词是指对于中文语句进行各个词的分隔，通常以语句为单位进行各个词的分离。一般来说，词性标注与分词系统由同一词法分析系统完成，这样能够保证分词过程和词性过程的良好衔接，如系统内所存储的词和相应词性有着良好的对应关系。

2.2分词应用代码及实例

2.2.1分词代码：



2.2.2实例：

①“也想要到人间去享一享这荣华富贵。”经过分词后变为“也/想要/到/人间/去/享一享/这/ 荣华富贵。”；

②“将一块大石登时变成一块鲜明莹洁的美玉”经过分词后变为“将/一块/大石/登时/变成/一块/鲜明/莹洁/的/美玉”；

③“然后好携你到那昌明隆盛之邦、诗礼簪缨之族、花柳繁华地、温柔富贵乡去安身乐业。”经过分词后变为“然后/好/携/你/到/那/昌明/隆盛/之邦/ 、/诗礼/簪缨/之族/ 、/花柳/繁华/地/ 、/温柔/富贵/乡去/安身乐业/。”；

④“原来女娲氏炼石补天之时，于大荒山无稽崖炼成高经十二丈、方经二十四丈顽石三万六千五百零一块。”经过分词后变为“原来/女娲/氏/炼石补天/之/时/，/于/大/荒山/无稽/崖/炼成/高经/十二/丈/、/方经/二十四丈/顽石/三万/六千五百/零/一块/。”。

通过上述实例可以看出，用jieba模块自带的分词词典可以解决如①和②这样的简单白话短语句和③这样的白话长语句，但是对于④这样的命名实体较多的长语句难以准确分词，这时就需要自己定义适当的词典来进行补充，对于《红楼梦》这一类的白话、古话混杂的语句需要自行定义词典才能进行准确的分析，jieba模块支持自定义扩展词典。

2.2.3导入自定义词典

（1）以dict.txt形式导入



该方法可以快速的将自己定义好的词典进行导入，同时可以定义词频和词性。

例如：贾母 4 n

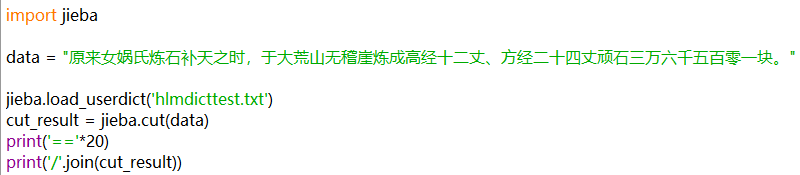
（2）以单个词导入



该方法用于快速将需要分词的词导入，少量词时可用该方法，同时支持词频和词性的导入。

（3）实例：

以2.2.2中④为例

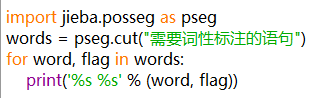


结果：原来/女娲氏/炼石补天/之/时/，/于/大荒山/无稽崖/炼成/高经十二丈/、/方经二十四丈/顽石/三万/六千五百/零/一块/。

1. **词性标注**

3.1词性标注应用代码及实例

3.1.1词性标注代码：



3.1.2实例：

①“你看看这玉上穿的穗子”：

你r看看v这r玉n上f穿v的uj穗子n

②“那贾母见他两个都生了气，只说趁今儿那边看戏，他两个见了也就完了，不想又都不去。”：

那 r 贾母nr 见v他r两个m都d生v了ul气n ，x只说c趁p今儿t那边r看v戏n，x他r两个m见v了ul也d就d完v了ul，x不想v又d都d不d去v 。x

③“一时宝钗、凤姐儿去了，黛玉笑向宝玉道："你也试着比我利害的人了。谁都像我心拙口笨的，由着人说呢！"宝玉正因宝钗多了心，自己没趣，又见黛玉来问着他，越发没好气起来。待要说两句，又恐黛玉多心，说不得忍着气，无精打彩一直出来了。”：

一m时宝钗nr、x凤姐儿nr去v了ul，x黛玉笑nr向宝玉nr道 q：x"x你r也d试v着uz比p我r利害v的uj人n了ul。x谁r都d像v我心n拙ag口q笨a的uj，x由着c人n说v呢y！x"x宝玉nr正d因p宝钗nr多m了ul心s，x自己r没趣v，x又d见v黛玉nr来v问v着uz他r，x越发d没好气l起来v。x待v要说c两句m，x又d恐d黛玉nr多心n，x说不得l忍v着uz气n，x无精打彩i一直d出来v了ul。x

标注句子分词后每个词的词性，采用和 ictclas 兼容的标记法，同时对标点符号进行了标记。jieba的词性标注也存在对古文标记不准确的问题，同样需要自定义词典来准确标记。

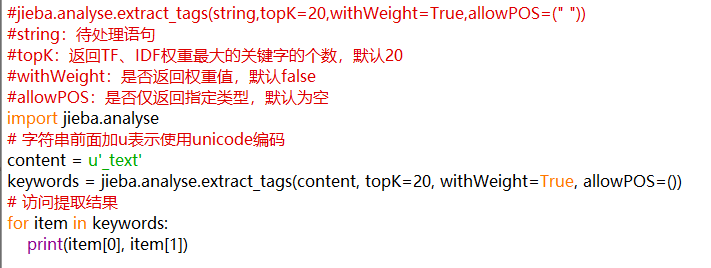
1. **基于TF-IDF算法的关键词提取**

4.1 TF-UDF已在第三部分文本分析的主要术语及其应用中详细解释。

4.2关键词提取代码及实例

4.2.1语句关键词提取

代码：



实例：

“原来贾珍近因居丧,每不得游顽旷荡,又不得观优闻乐作遣.无聊之极,便生了个破闷之法. 日间以习射为由,请了各世家弟兄及诸富贵亲友来较射.因说:"白白的只管乱射,终无裨益,不但不能长进,而且坏了式样,必须立个罚约,赌个利物,大家才有勉力之心."因此在天香楼下箭道内立了鹄子,皆约定每日早饭后来射鹄子.贾珍不肯出名, 便命贾蓉作局家.这些来的皆系世袭公子,人人家道丰富,且都在少年,正是斗鸡走狗,问柳评花的一干游荡纨裤.因此大家议定,每日轮流作晚饭之主,----每日来射, 不便独扰贾蓉一人之意.于是天天宰猪割羊,屠鹅戮鸭,好似临潼斗宝一般,都要卖弄自己家的好厨役好烹炮.不到半月工夫,贾赦贾政听见这般,不知就里,反说这才是正理, 文既误矣,武事当亦该习,况在武荫之属.两处遂也命贾环,贾琮,宝玉,贾兰四人于饭后过来,跟着贾珍习射一回,方许回去.”

关键词：由于文字较少，所以按权值占比选前五个

贾珍 0.2248850718385

每日 0.15875041022175002

问柳评花 0.11583898043333334

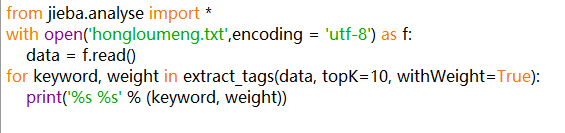
临潼斗宝 0.11583898043333334

纨裤 0.10428652742416666

4.2.2 文本关键词提取

实例：以红楼梦.txt为例，取权值占比前十个。

代码：



宝玉 0.11174015386450631

贾母 0.04096279485052852

凤姐 0.03679341393983838

王夫人 0.03238483108809296

老太太 0.029228577284745143

奶奶 0.025319232252404632

那里 0.024846723669342888

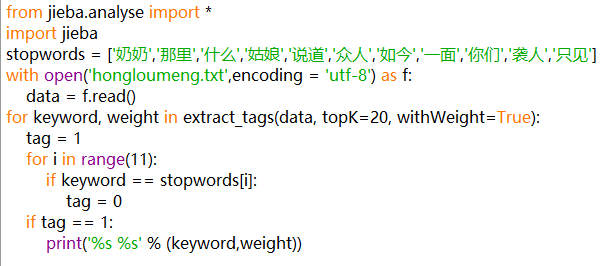
什么 0.024490410824605114

贾琏 0.02434048230267443

太太 0.024266427970154856

4.2.3 导入自定义停用词

自定义部分停用词后分析红楼梦文本，取权值前十个



结果：

宝玉 0.1117401538645063

贾母 0.0409627948505285

凤姐 0.03679341393983838

王夫人 0.03238483108809296

老太太 0.029228577284745143

贾琏 0.02434048230267443

太太 0.024266427970154856

平儿 0.02184055288925734

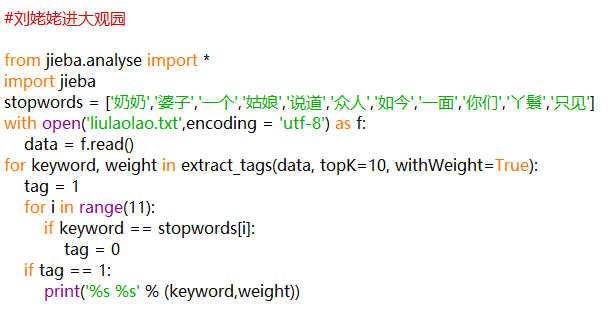
宝钗 0.01920189352608345

黛玉 0.018863309330642977

4.2.4 具体章节关键词提取

实例：以《刘姥姥进大观园》为例，运用TF-IDF进行关键词提取，选前十个。

代码：



刘姥姥 0.2506356606436061

鸳鸯 0.13411782426843366

贾母 0.12497564775518509

凤姐儿 0.06454866313749684

宝玉 0.05971277286262906

薛姨妈 0.05021407398835558

凤姐 0.045445349038360616

王夫人 0.04531095804545203

妙玉 0.04049571412863259

4.3 分析

通过对语句和文本的关键词提取可以明显的看出，对一般语句的关键词提取可以轻松的找出这段语句的中心含义，但对文本的关键词提取却很难分析中心含义，往往是通过自定义词典来得到想要的结果，比如《红楼梦》中人物的重要程度。TF-IDF关键词提取更适用于具体章节的分析，4.2.4实例的结果就非常清晰的找出了各个关键词。