

**首都师范大学本科实验设计论文**

小型搜索系统

Small search system

论文作者： 王润宇

院 系： 信息工程学院

专 业： 计算机科学与技术

学 号： 1161005202

指导老师： 宋巍

完成日期： 2020.7.2

**摘 要**

随着信息时代科技的不断进步，互联网上的信息不断膨胀，丰厚的网络资源为我们的日常生活带来了空前的便利，但与此同时，由于信息的爆炸性增长，极大的影响了用户获取信息的效率。为了解决这个问题，搜索系统便应运而生。

本文将会介绍如何基于文档来实现一个小型搜索系统，主要内容包括中文分词，倒排索引，评分函数，文本摘要，文本分类。通过这个小型搜索系统展现网络信息搜索系统的工作过程。

**关键词：**分词，TF-IDF，索引，线性模型，sklearn

**ABSTRACT**

With the continuous progress of science and technology in the information age, the information on the Internet continues to expand, rich network resources for our daily life has brought unprecedented convenience, but at the same time, due to the explosive growth of information, it greatly affects the efficiency of users to obtain information. In order to solve this problem, search system came into being.

This paper will introduce how to implement a small search system based on documents, including Chinese word segmentation, inverted index, scoring function, text summary, text classification. This small search system shows the working process of the network information search system.

**Key Words**: Word segmentation, TF-IDF, index, linear model, sklearn

目 录

[第一章 绪论 4](#_Toc26360)

[1.1 论文研究背景及意义 4](#_Toc5825)

[1.1.1 研究背景 4](#_Toc14040)

[1.1.2 研究意义 4](#_Toc30252)

[1.2 论文结构与内容安排 4](#_Toc31457)

[第二章 小型搜索系统的实现 5](#_Toc416270784)

[2.1 使用说明 5](#_Toc416270785)

[2.2 基于开源工具JIEBA的中文分词 6](#_Toc416270785)

[2.2.1 JIEBA概述 6](#_Toc416270786)

2.2.2 如何用JIEBA进行分词 6

[2.3 倒排索引 7](#_Toc416270788)

[2.3.1 倒排索引概述 7](#_Toc416270789)

[2.3.2 倒排索引的建立 7](#_Toc416270790)

[2.4 基于TF-IDF的评分函数 1](#_Toc416270791)0

[2.4.1 TF-IDF概述 1](#_Toc416270792)0

[2.4.2 用TF-IDF实现评分函数 1](#_Toc416270793)0

[2.5 文本摘要 1](#_Toc416270794)2

[2.5.1 文本摘要的原理 1](#_Toc416270795)2

[2.5.2 文本摘要的建立 1](#_Toc416270796)2

[2.6 基于开源工具sklearn的文本分类 1](#_Toc416270794)3

[2.6.1 文本分类概述 1](#_Toc416270795)3

2.6.2 文本分类的实现 13

[参 考 文 献 18](#_Toc416270797)

[致 谢 19](#_Toc416270799)

# 第一章 绪论

## 1.1 论文研究背景及意义

### **1.1.1 研究背景**

随着互联网的迅速发展，搜索引擎已成为人们必不可少的一部分。搜索引擎以其强大的抓取能力，抓取范围现阶段已经覆盖了全网百分之二十左右的网页。Google（谷歌）的网页覆盖量有60亿左右，百度也达到10亿。Google被誉为互联网最伟大的公司，百度成为中国第二大互联网公司，这都宣告着搜索时代的大发展，搜索引擎给在互联网遨游的人们带来了巨大的方便。

搜索引擎不再是单一的一搜一点，以百度为例，不在只是自然结果，各种结构化的展现结果出现了。例如搜一个人的名字，立马会给出百科、电影、图片、知道、微博、新闻等信息，这是建立在用户日志的基础上，给出的最佳搜索结果，其中的每一条结果，都明显好于简单的自然结果，这是众多产品人员和程序人员辛勤工作的结果。

这些都深刻的说明着搜索引擎的现阶段发展的普及性和多样性，搜索引擎的大发展时代已经来临，搜索引擎的研究对人们的互联网生活意义重大[1]。

### **1.1.2 研究意义**

本文主要介绍通过文档完成的一个小型搜索系统，以此来展现搜索引擎的工作原理，如中文分词，建立倒排索引来提高检索的效率，对搜索到的文章进行评分，有利于将用户希望找到的内容呈现给用户，提取摘要来减少用户读取文章的时间，文本分类等。

## 1.2 论文结构与内容安排

[搜索引擎](https://www.baidu.com/s?wd=%E6%90%9C%E7%B4%A2%E5%BC%95%E6%93%8E&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao" \t "https://zhidao.baidu.com/question/_blank)已经成为互联网[不可缺少](https://www.baidu.com/s?wd=%E4%B8%8D%E5%8F%AF%E7%BC%BA%E5%B0%91&tn=SE_PcZhidaonwhc_ngpagmjz&rsv_dl=gh_pc_zhidao" \t "https://zhidao.baidu.com/question/_blank)的工具，可以帮助人们更快的找到所要的内容和信息。提高做事的效率，使互联网资源高效的利用。实现小型搜索系统的任务大致可以分为以下几个子课题

1. 基于jieba的中文分词
2. 建立倒排索引
3. 基于tf-idf的评分函数
4. 基于关键词的文本摘要
5. 基于sklearn开源工具中线性模型的文本分类

中文分词的方法很多，有最大正向匹配分词，最大逆向匹配分词，基于隐马尔科夫模型的中文分词，但我选择了利用开源工具进行分词，因为它具有3种分词模式，自带含有2万多词条的字典，采用trie树结构进行存储，具有查找速度快的优势。

建立倒排索引可以提高检索文档的效率，通过建立词与文章之间的数据结构来减少系统在查找文档过程中的开销。

基于tf-idf的评分函数，主要是通过tf-idf方法计算关键词（即用户输入信息分词后得到的各个词项）的分数，以此来为找到的文档进行评分。

基于关键词的文本摘要，通过计算每句话中出现的关键词的tf-idf值来为文章中的每句话进行打分，以此来提取文本摘要。

基于sklearn开源工具中线性模型的文本分类，通过sklearn开源工具对文本分类数据进行线性模型的训练，其中，80%为训练数据，20%为测试数据。

# 小型搜索系统的实现

2.1 使用说明

由于我没有做前端，所以就没有采用django框架，因此我做的小型搜索系统只能在编辑器下面的窗口展现我做的成果，文本摘要那部分我在每句摘要的后面将分数也放了上去，这样可以直观地看到摘要中每个句子的重要程度，其中摘要的数量我设置为不超过5句，如果找到的摘要总数小于或等于5就全部展示出来。中文分词我是采用开源工具jieba来实现的，对找到的各个文章是通过TF-IDF来进行评分的。我是用sklearn中的线性模型来实现文本分类的，文本分类训练好的模型我用pickle开源工具存放在.pkl文件中，我把建好的倒排索引存放在了.txt文件中。由于建立倒排索引和文本分类的线性模型花费的时间有点长，我把它们两个的建立和存放到文件中的函数注释掉了，每次只需要调用从文件中载入倒排索引和线性模型的函数就可以了，能节约很多时间，运行大作业.py文件就可以使用这个小型搜索系统了。

## 2.2 基于开源工具JIEBA的中文分词

**2.2.1 JIEBA概述**

一、支持三种分词模式：

精确模式，试图将句子最精确地切开，适合文本分析；全模式，把句子中所有的可以成词的词语都扫描出来, 速度非常快，但是不能解决歧义；搜索引擎模式，在精确模式的基础上，对长词再次切分，提高召回率，适合用于搜索引擎分词。

1. jieba自带了一个叫做dict.txt的词典

词典里面有2万多条词, 包含了词条出现的次数(这个次数是于作者自己基于人民日报语料等资源训练得出来的)和词性. 这个第一条的trie树结构的词图扫描, 说的就是把这2万多条词语, 放到一个trie树中, 而trie树是有名的前缀树, 也就是说一个词语的前面几个字一样, 就表示他们具有相同的前缀, 就可以使用trie树来存储, 具有查找速度快的优势。

三、jieba分词属于[概率语言模型分词](http://book.51cto.com/art/201106/269050.htm" \t "https://www.cnblogs.com/echo-cheng/p/_blank)

概率语言模型分词的任务是：在全切分所得的所有结果中求某个切分方案S，使得P(S)最大[2]。

**2.2.1 如何用JIEBA进行分词**

首先在文件开头输入import jieba来调入jieba工具包,然后就可以通过调用函数jieba.lcut()来对一段字符串进行分词，由于这个函数不能将停用词过滤掉，所以我还需要从停用词文档中把所有的停用词提取出来以便于过滤分词结果。此外我还规定了词的长度不能小于2，因为我认为小于2的是字，不是词。代码如下(具体代码可到search\_engine包中的jieba\_cut.py中查看)。

import jieba

import jieba.analyse

def Jieba(sentence):

words = []

useless = []

with open('stoplist\_utf8.txt', encoding='utf-8') as fread:

for line in fread:

line = line.strip() # 去掉两端的空白字符，如空格、回车等

useless.append(line)

fread.close()

word = jieba.lcut(sentence)

for w in word:

# 排除停用词且长度不能太低

if (w not in useless and len(w) > 1):

words.append(w)

return words

2.3 倒排索引

**2.3.1 倒排索引概述**

倒排索引是搜索引擎中使用的主要数据结构。他允许快速查找包含任何给定词项的文档。相关的数据结构包括：

1. 词典（即查找表，包含词项特定信息，如文档频率和它在记录文件中的位置）
2. 记录文件（从任意词项的整数ID到一个文档ID列表的映射，记录词项在这些文档中出现的频率）。

**2.3.2 倒排索引的建立**

依次扫描所有文档。首先为每个文档分配一个ID。对文档分词从而为词项分配ID，并根据需要创建新词项ID。在扫描文档时，为每个文档-词项构成的对进行数量累计，记录每个文档中每个词的词频，以此来建立一个倒排索引。代码如下(具体代码可到search\_engine包中的invertedIndex.py中查看)。

2.4 基于TF-IDF的评分函数

**2.4.1 TF-IDF概述**

TF-IDF（Term Frequency-Invers Document Frequency）即为文档评率-逆文档频率。该技术采用一种统计方法，根据字词的在文本中出现的次数和在整个语料中出现的文档频率来计算一个字词在整个语料中的重要程度。它的优点是能过滤掉一些常见的却无关紧要本的词语，同时保留影响整个文本的重要字词。TF表示词条在文档d中出现的频率，IDF的主要思想是假设包含词条t的文档越少，也就是df(w)越小，IDF越大，则说明词条t具有很好的类别区分能力，且IDF对于高频词具有很好的打压能力。公式如下：

tf(w,d) = count(w,d) (2.1)

 (2.2)

 (2.3)

**2.4.2 用TF-IDF实现评分函数**

首先对用户输入的信息进行分词，然后根据分词得到的词的集合q通过倒排索引查找文档d，看d匹配了几个q中的关键词，并给每一个匹配的关键词分配一个分数，分数越高说明越相关。代码如下(具体代码可到search\_engine包中的score\_sort.py中查看)。

2.5 文本摘要

**2.5.1 文本摘要的原理**

因为用户更希望看到与他输入的信息更相关的那些句子作为摘要，所以通过计算每个文档中各个句子中关键词的分数之和来衡量这句话在文档中的重要程度，从而将最相关的句子提取出来作为文本摘要。

**2.5.2 文本摘要的建立**

通过用户输入的信息进行分词后得到的词的集合q找出候选文档中相应的句子，看这些句子匹配了几个q中的关键词，并给每一个匹配的关键词分配一个分数来求出句子的分数，以此来进行排序，最后提取排名靠前的句子作为摘要，代码如下(具体代码可到search\_engine包中的score\_sort.py中查看)。

2.6 基于开源工具sklearn的文本分类

**2.6.1 文本分类概述**

给定一组预先设定的类别，同时还存在一组带标签的文本对象（即已知类别标签的文本对象），文本分类的任务是给未知文本对象赋予一个或多个类别标签。文本分类可以丰富文本表示，如主题分类或情感分类，相当于给文本本身赋予更丰富的标签。

**2.6.2 文本分类的实现**

首先导入开源工具sklearn以及存储模型的工具包pickle，即在文件的开头输入调用相关工具包的代码（如import pickle,from sklearn...），然后对文本分类数据进行预处理操作，即将标签和文档分别整合到列表中（注:标签不能进行去重操作，文档的格式必须为字符串且词与词之间以空格分开，例texts=["dog cat fish", "dog catch cat"]）。接着调用TfidfVectorizer()和fit\_transform()函数生成稀疏矩阵，利用train\_test\_split()将文本分类数据分成训练集和测试集，其中test\_size为设置测试集所占比例，然后通过LinearSVC().fit()对划分好的训练集的数据进行建模，将模型建立好之后就可以用pickle.dump()把模型存放到指定路径的磁盘中，通过pickle.load()从磁盘中载入模型，训练好的模型可以通过sklearn中的score()函数对测试集上的数据进行打分，还可以用classification\_report()函数生成文本分类报告。代码如下(具体代码可到search\_engine包中的classification.py中查看)。

# 参 考 文 献

[1] http://www.698job.com/ArticleShow.asp?ArtID=1925，搜索引擎发展的背景及意义，作者：花都网站建设

[2] <https://www.cnblogs.com/echo-cheng/p/7967221.html>，[jieba分词的原理 - hello\_echo -博客园](https://www.baidu.com/link?url=0WZCk0b13EEjLQKLgX-76ZblkbfMP8ZghmZUi8Xnt5VXz9xlRP8E0JxTUVd83pVgZgrNoT8SsXaa-CgSyCXIR_&wd=&eqid=961f2ca700000456000000025efc4aa3" \t "https://www.baidu.com/_blank)

# 致 谢

这次大作业让我受益匪浅，我不仅掌握了python这门语言，还对机器学习和自然语言处理有了更深刻的了解。

在大作业设计的过程中，我学习到了很多python的数据结构和开源工具。首先，我学会了如何使用list列表和dict字典，我还掌握了python中面向对象的模块化编程的设计理念，利用模块化编程将复杂的系统转换为各个子模块，然后逐个击破，使复杂的问题变得更加简单直观。其次，我在构建搜索系统的过程中，学到很多基础理论知识，如隐马尔科夫模型，knn算法，贝叶斯定理，线性模型，TF-IDF等。同时，这也是我第一次接触机器学习，这个过程是非常艰难的，我通过到网上看视频和查阅资料以及不断的尝试和修改代码，渐渐的理解了如何利用开源工具进行机器学习来完成建模。

尽管这学期由于疫情原因，我无法和宋巍老师一同坐在教室中探讨这门课的内容，但是宋巍老师通过直播的形式让我通过这门课学习到了很多知识。此外，在批改完作业后还在批语中告诉我如何优化代码，极大的提高了我的代码质量，在此我要衷心的感谢宋巍老师的悉心教导。