2020-2021年度山东大学软件学院信息检索考试

开学的时候上这门课时老师说很多同学都觉得这门课程难（范围广且多，不过这倒是符合信息检索的定义“融合了~~~的综合学科”，哈哈哈），最终期末考试是“贝多芬（背多分）”，纯记忆力考试。写这个回忆版也耗费了我巨大的记忆力，我如何记得住这么多题（无它，答完后还剩50分钟，背就完了），如果我考前不喝红牛或许还记不住这么多题。复习的时候获得了学长的回忆版，可惜没有填空题，传承“传统”也就有了这个回忆版，话不多说，开始我的回忆……

考试时间：2020年12月22日 14:00-16:00

保证所有题目顺序的正确性

去年题（2019-2020）：https://download.csdn.net/download/qq\_40422851/12054761

一、填空题（每空1分，共38分）

信息检索实质上融合了文本及多媒体检索、数据挖掘、\_\_机器学习\_\_\_ 和 \_\_自然语言处理\_\_\_\_ 的综合学科。

信息检索技术可以解决互联网上的 \_\_信息量太大\_\_\_ 问题。

爬虫是一种 \_\_自动获取网页内容\_\_\_ 程序。

爬虫从一个或若干个网页的URL开始，抽取URL放入 \_\_队列\_\_\_ 中。

爬虫可以采用 \_\_哈希法\_\_\_ 判重。

爬虫的搜索策略有 \_\_深度优先\_\_\_ 和 \_\_广度优先\_\_\_ 。

爬虫礼貌性体现在 \_\_\_\_\_\_\_Web服务器有显示或隐式的策略控制爬虫的访问\_\_\_\_\_\_\_\_\_\_ 。

使用python语言中的Pandas模块里的 \_\_\_lxml\_\_ 函数，可以直接提取网页中的表格。

搜索引擎的主要组成部分为：爬虫和爬取、\_\_建立索引\_\_\_ 、搜索词处理、排序。搜索引擎的核心数据结构为 \_\_倒排索引\_\_\_ 。

消除停用词的方法： \_\_查表法\_\_\_ 和 \_\_\_基于文档频率\_\_。

\_\_\_词干还原\_\_ 在一般情况下，将多个派生词合并在一起。

基于词典的分词方法，也叫“ \_\_\_查字典\_\_ ” 法，根据扫描方向，匹配策略有 \_\_正向匹配/逆向匹配\_\_\_ 和 \_\_\_最大匹配/最小匹配\_\_ 。

布尔模型建立在 \_\_\_经典的集合论\_\_ 和 \_\_\_布尔代数\_\_ 的基础上。

精确的Top K搜索的加速方法有（1）\_\_快速计算余弦\_\_\_ ； （2） \_\_\_堆排序法N中选K\_\_ ；（3）\_\_\_提前终止计算\_\_ 。

L2R中的文档方法损失函数评估 \_\_\_\_\_ 。

PageRank是基于 \_\_“从许多优质的网页链接过来的网页，必定还是优质网页”\_\_\_\_ 的回归关系来判定所有网页的重要性。

HITS算法中，每个网页计算两个值：\_\_\_权威值\_\_ 和 \_\_中心值\_\_\_ 。

SVD也就是 \_\_奇异值分解\_\_\_ ，可以用于 \_\_矩阵分解\_\_\_ 。

BM25由三个部分组成： \_\_query中每个单词t与文档d之间的相关性\_\_\_ 、\_\_单词t与query之间的相似性\_\_\_ 和 \_\_每个单词的权重\_\_\_ 。

词的向量表示有 \_\_\_\_\_ 和 \_\_\_\_\_ 。

用于大量文档查重的局部敏感哈希算法有 \_\_\_\_\_ 和 \_\_\_\_\_ 。

形状的描述符大体可以分为两大类，基于 \_\_\_\_\_ 的和基于 \_\_\_\_\_ 的。

二、名词解释（每个6分，共30分）

正则表达式

图像BoW

统计语言模型

pLSA

LBR

三、简答题（每个8分，共32分）

简述向量空间模型及其特点

什么是HMM？简述HMM用于中文分词的方法

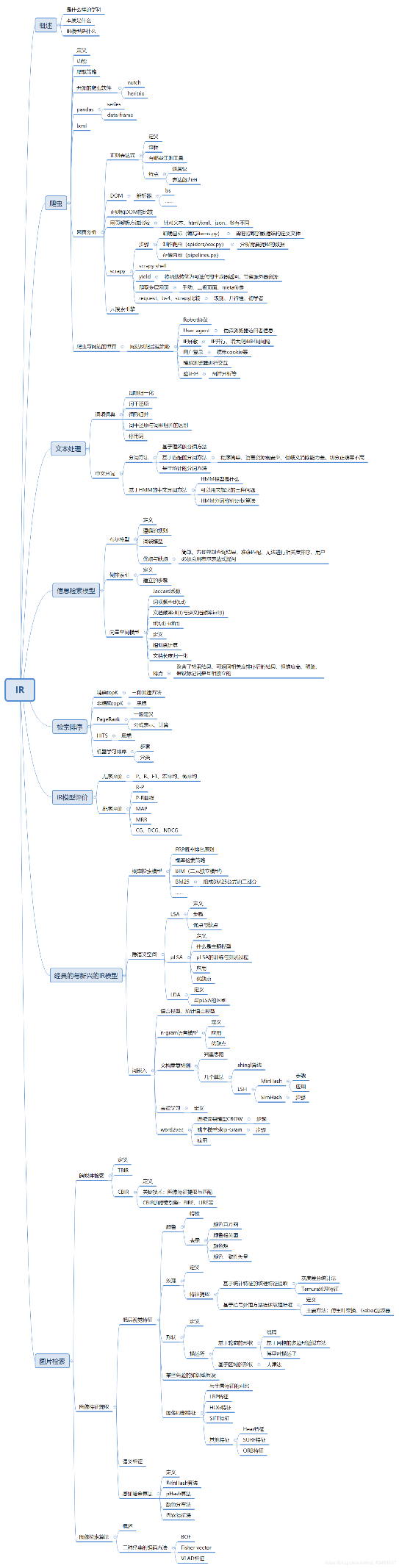
简述信息检索评价的主要指标

网站反爬虫的方法有哪些？应对策略是什么？

————————————————

版权声明：本文为CSDN博主「ALTLI」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：<https://blog.csdn.net/weixin_43360801/article/details/111561085>



第一讲 搜索

IR（信息检索是什么样的学科）

实质上是融合了文本及多媒体检索、数据挖掘、机器学习和自然语言处理的综合学科

为什么要进行信息检索？信息过载

搜索

搜索的过程

从大规模非结构化数据(通常是文本)的集合(通常保存在计算机上)中找出满足用户信息需求的资料(通常是文档)的过程

信息检索的本质

确定文档和查询之间的相关度是IR的核心问题

IR作为一门学科，是研究信息的获取(acquisition)、表示(representation)、存储(storage)、组织(organization)和访问(access)的一门学问

信息检索本质：给定一个查询Q，从文档集合C中，计算每篇文档D与Q的相关度，并排序(Ranking)

什么是相关度

相关度是一个查询和文档相关的程度，形式上说，信息检索中的相关度是一个\*\*函数\*f\*，\*\*输入是查询Q、文档D和文档集合C，返回的是一个实数值 R, R = f(Q,D,C)

相关度(relevance)不同于相似度(Similarity):

​ 相关度通常只有相对意义

​ (1)相关取决于用户的判断，是一个主观概念

​ (2)不同用户做出的判断很难保证一致

​ (3)即使是同一用户在不同时期、不同环境下做出的判断也不尽相同

定义“相关性”的两个角度：(了解)

系统角度：系统输出结果，用户是信息的接受者。

用户角度:观察用户对检索结果的反应，是系统输出向用户需求的投射

现代信息检索研究中仍然主要采用系统角度定义的主题相关性概念，当然也强调考虑用户的认知因素

信息检索模型

描述信息检索中的文档、查询和它们之间关系(匹配函数)的数学模型

信息检索主要技术

（1）文本分析（NLP）

（2）建立索引

（3）查询，包括查询分析（NLP），相关度计算(和信息检索模型相关)

（4）排序（实验室评价）

搜索引擎

工作原理

（1） 爬行和抓取

（2） 文本分析

（3）建立索引（可能会考的知识点：蜘蛛抓取的页面文件分解、分析，并以巨大表格的形式存入数据库，这个过程即是索引（index).搜索引擎的核心数据结构为倒排文件（也称倒排索引））

（4）搜索词处理 （5）排序 （6）用户反馈

搜索引擎评价

（1） 覆盖面 （2）更新周期 （3）响应速度 （4）排序结果是否满足用户的查询要求

第二讲 网络爬虫技术

爬虫定义

一种自动获取网页内容的程序，从一个或若干初始网页的\*\*URL开始，获取并解析它们，提取它们指向的URL，将提取的url放在队列中，获取队列中的每个URL并重复此过程，直到满足系统的一定停止条件\*\*

通俗的讲，也就是通过HTML源码解析来获得想要的内容

爬虫必须具有的功能

4.1 礼貌性: Web服务器有显式或隐式的策略控制爬虫的访问

只爬允许爬的内容、尊重 robots.txt

4.2 鲁棒性: 能从采集器陷阱中跳出，能处理Web服务器的其他恶意行为

4.3 性能和效率: 充分利用不同的系统资源，包括处理器、存储器和网络带宽

优先抓取“有用的网页”

4.4 分布式: 可以在多台机器上分布式运行

​ •分布式带来的问题

​ –哈希表判重

​ •解决方法：

​ –A、明确每台下载服务器的分工，即一看到某个URL就知道交给哪台服务器去执行

​ –B、批量处理，减少通信的次数

可扩展性: 添加更多机器后采集率应该提高

4.5 新鲜度: 对原来抓取的网页进行更新

4.6功能可扩展性：支持多方面的功能扩展，例如处理新的数据格式、新的抓取协议等

爬取框架

3、搜索策略：深度优先, 广度优先

​ 实际应用的网络爬虫不是对网页次序的简单BFS或者BFS，而是一个相对复杂的下载优先级排序的方法，管理这个系统的叫做“调度系统”(Scheduler)，会有一个Priority Queue。BFS成分更加多一些。

4、URL 判重

建立一个散列，其中存放访问过每一个网址

在其中存放网址经过散列函数计算出的对应的固定长度的散列值

在平均情况下\*\*O(1)\*\*的时间内查找和更新占用O(n)空间的网址列表

利用哈希法，URL经过哈希函数得到哈希码，判断是否已经在散列中来判断是否爬取过

爬虫分类

•5.1基于整个Web的信息采集(Universal Web Crawling)

​ •传统的采集方式

​ –作为门户搜索引擎和大型的Web服务提供商的数据收集部分

​ –是指从一些种子URL扩充到整个Web的信息采集

•5.2 增量式Web信息采集 (Incremental Web Crawling )

•5.3 基于主题的Web信息采集(Focused Web Crawling )

•5.4 基于用户个性化的Web信息采集(Customized Web Crawling )

•基于元搜索的信息采集(Metasearch Web Crawling)

常见的开源爬虫

Nutch Heritrix

•包括全文搜索和Web爬虫。

​ –包括爬虫crawler和查询searcher。

​ •Crawler主要用于从网络上抓取网页并为这些网页建立索引。

Pandas模块

lxml模块

•lxml是一个HTML/XML的解析库

•主要功能是如何解析和提取HTML/XML数据

第三讲 网页分析技术

网页解析方法

–一种是将文档看作字符流;

•正则表达式

–一种是将文档看作树结构。

•基于DOM

正则表达式

1、正则表达式的定义

正则表达式是对\*\*字符串操作的一种逻辑公式，就是用事先定义好的一些特定字符、及这些特定字符的组合，组成一个“规则字符串”，这个“规则字符串”用来表达对字符串的一种过滤逻辑。\*\*

2、基于正则表达式的信息提取的步骤

（1）在获取数据前应尽量去除无用部分（2）提取网页内的链接 （3）提取网页标题（4）提取网页内的文本

3、正则表达式的工具有哪些

Java java.util.regex包 Python的 re模块

4、正则表达式匹配特点是什么

（1）正则表达式匹配速度快，

（2）但表达能力较弱，只具有正规文法的表示能力。

（3）在对网页内容的信噪比要求不高的情况下可以使用基于正则表达式匹配的爬取程序

（4）受网页噪音影响较大

DOM

5、什么叫做DOM

文档对象模型（document object model，DOM）,DOM将一个XML文档转换成一个对象集合，然后可以任意处理该对象模型。

DOM将HTML视为树状结构的元素，所有元素以及他们的文字和属性可通过DOM树来操作与访问。

6、开源HTML解析器（能够列出一两种即可）

（1）JAVA：HTMLParser,jsoup

（2）C/C++：htmlcxx

（3）Python：Beautiful Soup

bs 解析器

–使用自带的html.parser解析，

​ •速度慢但通用

​ •soup = BeautifulSoup(html, “html.parser”)

–Html5lib

​ •不规范的html文本转为规范的文本再进行解析

​ 用浏览器的方式解析文档

–lxml

​ •python的一个解析库，

​ •支持HTML和XML的解析，

​ •支持XPath解析方式，

​ •而且解析效率非常高

​ •lxml只会局部遍历

两种方法比较

正则表达式匹配

（1）正则表达式匹配速度快，但表达能力较弱，只具有正规文法的表示能力。

（2）在对网页内容的信噪比要求不高的情况下可以使用基于正则表达式匹配的爬取程序

HTML DOM树

（1）提取HTML DOM树提取在解析HTML时速度较慢，但其表达能力相当于上下文无关文法。

（2）在网页自动分类等需要进行网页去噪处理的情况时使用基HTMLDOM树的爬取程序

Python爬虫

•工作过程

–把URL地址中指定的网络资源从网络流中读取出来，保存到本地

–过滤

​ Re

​ bs4

Scrapy shell

交互终端，不启动爬虫的情况下调试代码

直接用来测试XPath或者CSS表达式，不用import响应模块

查看运行的结果方便分析网页，测试表达式是否获取到了数据

python爬虫框架 Scrapy

•快速、高层次的屏幕抓取和web抓取框架，

•用于抓取web站点并从页面中提取结构化的数据。

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-2rmF6m42-1608430839949)(C:\Users\yandalao\AppData\Roaming\Typora\typora-user-images\image-20201216162520302.png)]

•爬虫文件novel\_spider.py

–分析需要提取的数据

​ •在parse方法中做数据的提取

​ •使用Xpath，从页面的HTML Source里面选取要要抽取的数据

Xpath

•XML路径语言（XML Path Language），它是一种用来确定XML文档中某部分位置的语言

•XPath基于XML的树状结构，提供在数据结构树中找寻节点的能力。

•xpath为scrapy中的解析方式

•xpath函数返回的为列表

​ –列表中存放的数据为Selector类型数据。

​ –解析到的内容被封装在Selector对象中，需要调用extract()函数将解析的内容从Selector中取出

Scrapy项目

•制作 Scrapy 爬虫 一共需要四步：

–新建项目 ：新建一个新的爬虫项目

–明确目标 （编写items.py）：明确你想要抓取的目标

​ •items.py: 需要提取的数据结构定义文件

​ –Item 定义结构化数据字段，用来保存爬取到的数据，

​ •修改novel\_spider.py : 分析需要提取的数据

–制作爬虫 （spiders/xxspider.py）：制作爬虫开始爬取网页

–存储内容 （pipelines.py）：设计管道存储爬取内容

yield

•只要是数据持久化存储，parse方法必须有返回值（也就是return后的内容）

​ –return items

•yield将函数转换成生成器。我们可以理解成一种特殊的return方法。

•yield返回的是一个生成器，也是可迭代对象，有利于减小服务器资源

•生成器相当于一种方法而不是具体的信息，占用内存小。

爬取多个网页

•start\_urls

•起始爬取列表，可以是多个url

start\_urls = (‘http://example.com/page1’, ‘http://example.com/page2’,)

爬取多层网页

•解析函数的末尾，通过Request方法对下一个页面手动发起请求

•\*\*先提取二级页面url，\*\*再对二级页面发送请求

比较

•request和bs4

–页面级爬虫，功能库

–并行性考虑不足，性能较差

–重点在于页面下载

•Scrapy

–网站级爬虫，框架

–并行性好，性能较高

–重点在于爬虫结构

元搜索引擎

•元搜索引擎又称多搜索引擎

•通过一个统一的用户界面帮助用户在多个搜索引擎中选择和利用合适的（甚至是同时利用若干个）搜索引擎来实现检索操作，是对分布于网络的多种检索工具的全局控制机制

第四讲 爬虫与网站的博弈

本章知道每个方面的思路和所用工具就可

Robot 协议

•网站通过Robots协议告诉搜索引擎哪些页面可以抓取，哪些页面不能抓取。

User-agent

•向访问网站提供访问者信息

•UA字符串在每次浏览器 HTTP 请求时发送到服务器！

–反爬虫

IP屏蔽

–爬虫：对策

•连接代理服务器

–写了个IP代理池

•多个IP并行

• 增大爬取时间间隔

用户登陆

•分析登陆过程的方法

•4.1 发送post请求

•4.2 分析post过程中隐藏的变量名

•4.3 分析 Cookie

​ –http 请求带着Cookie

​ •它记录了你的用户ID，密码、浏览过的网页、停留的时间等信息，用于用户身份的辨别

•流程

​ –\*\*第一个网页通过GET（\*\*\*\*POST)参数提交参数

​ •参数序列化成字符串

​ •和基础\*\*\*\*url 拼接

​ •Urllib.request.urlopen\*\*()\*\*

​ –后台接受请求，生成cookie，发给用户

​ –用户带着Cookie继续访问其他网页

•4.4 携带Cookie访问已登陆网站

•保存cookie到文件

•从文件中读取cookie并访问

•利用cookie模拟登录

模拟浏览器进行交互

selenium

•反爬虫： 用户登陆

–输入用户名

–输入口令

–点击登陆按钮

•Selenium用程序模拟整个操作过程

–忽略post或者get方式差异

–不需要知道参数名字

处理Cookie：

•selenium 获取登录\*\*\*\*cookies，

​ –selenium有一个 get\_cookies() 函数可以帮我们获取当前网页的cookie值

•保存cookies到文件

•并添加cookies自动登录

AJAX 动态加载

•通过在后台与服务器进行少量数据交换，AJAX 可以使网页实现异步更新。

在不重新加载整个网页的情况下，对网页的某部分进行更新

验证码

•图像识别

–6.1 获取图片

•分析网页下载图片

•屏幕截图

–6.2 图片处理 Pillow与PIL模块

–6.3 获取图片中文字内容 ocr

-6.4 图片滑动验证码

第五讲 词项词典

如何建立词项词典？

一、文档解析(Parsing a document)

~~二、词条化 (Tokenization)~~这俩不考

三、词项归一化 (Normalization)

四、词干还原 (Stemming)

五、词形归并 (Lemmatization)

六、去掉停用词 (Stop Words)

词项归一化

将文档和查询中的词条“归一化”成一致的形式（希望USA和U.S.A.之间也能形成匹配 ）

归一化的结果： 在IR系统的词项词典中，形成多个近似词项的一个等价类

策略：建立同义词扩展表

a) 为每个查询维护一张包含多个词的查询扩展词表

b) 在建立索引建构时就对词进行扩展

词干还原

a) 通常指去除单词两端词缀的启发式过程

b) 词干还原能够提高召回率，但是会降低准确率

词形归并

a) 利用词汇表和词形分析来减少屈折变化的形式，将其转变为基本形式。

b) 词形归并可以减少词项词典中的词项数量

词干还原和词形归并的区别

a) 代表意义不同。

​ i. Stemming通常指很粗略的去除单词两端词缀的启发式过程。

​ ii. Lemmatization通常指利用词汇表和词形分析来去除屈折词缀，从而返回词的原形或词典中的词的过程。

b) 两个过程的区别还在于：

​ i. 词干还原在一般情况下会将多个派生相关词合并在一起，

​ ii. 而词形归并通常只将同一词元的不同屈折形式进行合并。

c) 词干还原和词形归并，都体现了不同语言之间的差异性

d) 词干还原过程可能仅返回 s，

e) 而词形归并过程将返回see或者saw，

停用词

a) 应用太广泛，区分度太低

b) 对这样的词搜索引擎无法保证能够给出真正相关的搜索结果，难以帮助缩小搜索范围，同时还会降低搜索的效率

消除停用词的优缺点

a) 优点：

​ i. 停用词消除可以减少term的个数

​ ii. 缩小搜索范围，

​ iii. 提高搜索的效率

​ iv. 机器学习文本分类算法的文档的预处理

b) 缺点：

​ i. 有时消除的停用词对检索是有意义的

如何确定停用词

a) 查表法

b) 基于文档频率

第六讲 中文分词

分词方法

a) 基于理解的分词方法

NLP、语义分析、句法分析

b) 基于字符串匹配的分词方法

查字典。

按照扫描方向：正向匹配和逆向匹配

按照扫描长度：最大匹配和最小匹配

a) 优点：简单，占用资源少，可自定义词库

​ i. 程序简单易行，开发周期短；

​ ii. 仅需很少的语言资源（词表），

​ iii. 不需要任何词法、句法、语义资源。

​ iv. 可以自定义词库，增加新词

b) 缺点 ： 效果差

​ i. Out of Vocabulary

​ ii. 歧义消解能力差；

​ iii. 切分正确率不高，一般在95%左右。

c) 基于统计的分词方法

用字与字相邻出现的频率来反应成词的可靠度，统计语料中相邻出现的各个字的组合的频度，当组合频度高于某一个临界值时，我们便可认为此字组可能构成一个词语。

基于统计的分词方法的优缺点：

a) 优点：

​ i. 分词准确度高；

​ ii. 能够平衡地看待词表词和未登录词的识别问题。

b) 缺点：

​ i. 局限性，会经常抽出一些共现频度高、但并不是词的常用字组

​ ii. 对常用词的识别精度差，时空开销大

​ iii. 学习算法的复杂度往往较高，计算代价较大，依赖手工定义的特征工程

基于HMM的中文分词方法

HMM作用

用来描述一个含有隐含未知参数的马尔可夫过程。

隐含状态之间存在转换概率；隐含状态和可见状态之间存在发射概率

HMM模型是一个五元组:

StatusSet: 状态值集合

ObservedSet: 观察值集合

TransProbMatrix: 转移概率矩阵 A

EmitProbMatrix: 发射概率矩阵 B

–在某一状态下对应到某字的概率

–P(Observed[i]|Status[j])

•基于观察值只取决于当前状态值这一假设

•其实也是一个条件概率

InitStatus: 初始状态分布

​ –句子的第一个字属于{B,E,M,S}这四种状态的概率

•HMM三要素[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-ZlhDCqDG-1608430839951)(image\image-20201216190517905.png)]

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-BROKijaw-1608430839953)(image\image-20201216190525015.png)]

HMM模型可以用来解决三种问题

a) 模型参数学习问题

b) 预测问题

c) 评估观察序列概率

HMM分词

预测问题，也叫解码问题

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-NGSEDXN9-1608430839955)(image\image-20201216190642734.png)]

Viterbi 算法

如何分词：将句子中的词看成有可能四个状态BMES，最后求出最有可能的状态序列（根据路径）。就分词成功

一种动态规划算法，它用于寻找最有可能产生 观测事件 序列的维特比路径——隐含状态序列

•二维数组 weight[4] [7]

​ –4是状态数(0:B,1:E,2:M,3:S)，

​ –7是输入句子的字数。

​ –P(Observed[i]|Status[j])

​ »比如 weight[0] [2] 代表 状态B的条件下，出现‘市’这个字的可能性。

•二维数组 path[4] [15]

–path[0] [2] 代表 weight[0] [2]取到最大时，前一个字的状态，

•比如 path[0] [2] = 1, 则代表 weight[0] [2]取到最大时，前一个字(也就是明)的状态是E。

第七讲 布尔模型与倒排索引

1、什么是信息检索模型

信息检索模型（IR model），依照用户查询，对文档集合进行相关排序的一组前提假设和算法。IR模型可形式地表示为一个四元组< D, Q, F, R(qi,dj) >

D是一个文档集合，Q是一个查询集合，R(qi,dj) 是一个排序函数，它给查询qi和文档 dj 之间的相关度赋予一个排序值，F是一个框架,用以构建文档,查询以及它们之间关系的模型

2、基于内容的信息检索模型有哪些？

• 集合论模型：布尔模型、模糊集合模型、扩展布尔模型

• 代数模型： 向量空间模型、广义向量空间模型、潜在语义标引模型、神经网络模型

• 概率模型： 经典概率论模型、推理网络模型、置信（信念）网络模型

• 深度学习模型

3、布尔模型是什么

一种简单的检索模型，建立在经典的集合论和布尔代数的基础上

遵循两条基本规则:

（1）每个索引词在一篇文档中只有两种状态：出现或不出现，对应权值为 0或1。

（2）每篇文档：索引词（0或1）的集合

进行查询的时候，用布尔表达式进行匹配，计算二值的相关度。

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-Py4ldaW5-1608430839958)(image\image-20201217120733627.png)]

4、什么是bag of words 模型

在信息检索中，Bag of words model假定

（1）对于一个文本，忽略其词序和语法，句法，将其仅仅看做是一个词集合，或者说是词的一个组合，

（2）文本中每个词的出现都是独立的，不依赖于其他词是否出现，在任意一个位置选择一个词汇都不受前面句子的影响而独立选择的。

5、搜索引擎的核心数据结构为倒排文件（Inverted Files）（也叫倒排索引）

6、什么是倒排索引

有词项和倒排记录组成，\*\*词项词典：\*\*对于每一个词项，存储所有包含这个词项的文档的一个列表。\*\*倒排记录表：\*\*一个文档用一个序列号docID来表示。

•建立索引的步骤：

–词条序列Token Sequence

​ •(修改过的词条，文档ID)对 序列

–排序

​ •先按照词条排序，

​ •再按照docID排序

–构建词典和倒排表

​ •同一篇文档中多次出现的词被合并

​ •分割成词典和倒排表

9、布尔检索模型的特点是什么

优点：（1）查询简单，因此容易理解（下面的具体说明理解即可）

• 布尔模型也许是IR系统中的最简单的模型

• 是近30年来最主要的商业搜索工具

• 当前使用的很多系统依然是使用的布尔模型

• 电子邮件，图书馆分类系统，mac osx的spotlight

（2）通过使用复杂的布尔表达式，可方便地控制查询结果

• 同义关系 电脑 OR 计算机

• 词组 数据 AND 挖掘

缺点 （1）准确匹配，信息需求的能力表达不足。不能输出部分匹配的情况

（2）无权重设计 无法排序，

（3）用户必须会用布尔表达式提问，一般而言，检出的文档或者太多或者太少。

（4） 很难进行自动的相关反馈

第八讲 向量空间模型

排序检索

系统根据文档与query的相关性排序返回文档集合中的文档；有布尔查询和自由文本查询两种方式

Jaccard 系数

• 一种常用的衡量两个集合A,B重叠度的方法

• Jaccard(A,B) = |A ∩ B| / |A ∪ B|（回答这个公式即可）

• Jaccard(A,A) = 1

• Jaccard(A,B) = 0 if A ∩ B = 0

• 集合A和B不需要具有同样的规模

–没有考虑

​ •文档长短

​ •词项频率(词项在文档中出现的次数)

​ •罕见词比高频词的信息量更大，更加具有区分度

词项频率

词项t在文档d中出现的次数，记为tft,d）

一种替代原始tf的方法: 对数词频 原始的词频tf以10为底取对数再加一

什么是idf：是逆文档频率，idft = log10(N/dft)，df是文档频率，指出现词项的文档数目

文档频率 (Document frequency，df)

• 文档频率:出现词项的文档数目

• dft 文档集合中包含t的文档数目

– 与词项t包含的信息量成反比

– dft <= N (N是文档的总数)

idf (inverse document frequency)逆文档频率

• idft = log10(N/dft)

– idft 是反映词项t的信息量的一个指标

– 用log (N/dft) 代替N/dft 来抑制idf的作用

tf-idf是什么

是信息检索中最著名的权重计算方法，表示t对于文档d的重要程度，词项t的tf-idf 由它的tf和idf组合而成 wt,d=(1+log tft,d) × log10(N/dft)

（理解一下和重要程度是否符合：tf-idf值随着词项在单个文档中出现次数(tf)增加而增大，tf-idf值随着词项在文档集中数目(df)增加而减小）

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-s9lj0KLn-1608430839959)(image\image-20201217145033660.png)]

向量空间模型

是一个\*\*|V|维实向量空间\*\*（V是词项集合，|V|表示词项个数），空间的每一维都对应一个词项，每篇文档表示成一个基于tf-idf权重的实值向量，向量的维度是词项的个数，文档是空间中的点或者向量，这就是向量空间模型

向量相似度计算

余玄相似度：（认为cos(di,q) > cos(dj,q)，夹角更小，所以di比dj与q更相关）

R(d,q) = cos(d,q) = d·q/|d|×|q|

文档长度归一化

•一个文档向量除以它的L2 范数（Xi的平方和取根号）就是给这个文档进行了长度归一化

向量空间模型特点

优点:

（1）帮助改善了检索结果。

（2）部分匹配的文档也可以被检索到。

（3）可以基于向量cosine 的值进行排序，提供给用户。

缺点:

（1）这种方法假设标记词是相互独立的，但实际可能不是这样，如同义词、近义词等往往被认为是不相关的词

（2）维度非常高：特别是互联网搜索引擎，空间可能达到千万维或更高

（3）向量空间非常稀疏：对每个向量来说大部分都是0

第九讲 检索排序

精确top K 检索及其加速办法

(一般)步骤：对每个文档评分(余弦相似度)，按照评分高低排序，选出前K个结果

如何加速：

方法一：快速计算余弦

方法二：堆排序法N中选K（不对所有文档的评分结果排序而直接选出Top K篇文档）只是缩减了排序这一步骤

方法三：提前终止计算 （不需要计算所有Ｎ篇文档的得分

非精确top K检索

简答题不用细答，看看了解

基本思想：找一个文档集合A，K < |A |<< N，利用A中的top K结果代替整个文档集的top K结果

下面的策略就是为了缩减文档的数量

• 策略一：索引去除(Index elimination)

​ 只考虑那些词项的idf 值超过一定阈值的文档

​ 只考虑包含多个查询词项

• 策略二：胜者表(Champion list) 每个词项t对应tf值高的表

• 策略三：静态得分 不仅相关，还权威，根据相关和权威度加权，对doc进行排序

• 策略四：影响度(Impact)排序 以词项为单位，串行遍历词项的倒排索引表

• 策略五：簇剪枝方法—预处理

Pagerank算法

•随机游走模型 是个一阶马尔可夫链

​ –用来描述不稳定的移动。

​ –移动节点随机选择一个方向和速度来从当前位置移动到新的位置

PageRank的思路：在随机游走过程中访问越频繁的网页越重要

PageRank的一般定义

•PageRank一般定义的想法是在基本定义的基础上导入平滑项

一个一定平稳分布的马尔可夫链：

​ M是转移矩阵，–R 是n维向量，表示的就是有向图的一般PageRank

​ R = d M R + 1 − d n 1 R=d M R+\frac{1-d}{n} 1R=dMR+

n

1−d

​

1

​ •第一项表示（状态分布是平稳分布时）依照转移矩阵M访问各个结点的概率，

​ •第二项表示完全随机访问各个结点的概率

第一项表示：•在任意一个网页上，浏览者或者以概率d决定按照超链接随机跳转，这时以等概率从连接出去的超链接跳转到下一个网页

第二项表示：•或者以概率（1-d)决定完全随机跳转，这时以等概率1/n跳转到任意一个网页

•第二个机制保证从没有连接出去的超链接的网页也可以跳转出。这样可以保证平稳分布，即一般PageRank的存在，因而一般PageRank适用于任何结构的网络。

对于一个节点A

​ P R ( A ) = ( P R ( B ) L ( B ) + P R ( C ) L ( C ) + P R ( D ) L ( D ) + ⋯ ⋅ ⋅ ) d + 1 − d N P R(A)=\left(\frac{P R(B)}{L(B)}+\frac{P R(C)}{L(C)}+\frac{P R(D)}{L(D)}+\cdots \cdot \cdot\right) d+\frac{1-d}{N}PR(A)=(

L(B)

PR(B)

​

+

L(C)

PR(C)

​

+

L(D)

PR(D)

​

+⋯⋅⋅)d+

N

1−d

​

其中，PR(A)表示页面A的级别，页面Ti链向页面A，L(Ti) 是页面Ti 链出的链接数量

迭代算法

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-CgRIEJHX-1608430839960)(image\image-20201217155401700.png)]

HITS算法

了解思想就行

• 在HITS算法中,对每个网页都要计算两个值\*\*:权威值(authority)与中心值(hub)\*\*

HITS和PageRank的区别

a.HITS算法将重要性分为两个值权威值(authority)与中心值(hub)，PageRank只计算一个值

b.HITS和查询有关系，PageRank算法和查询无关

机器学习排序

步骤：

–人工标注训练数据，给出文档和查询相关度

–文档特征抽取、确定特征数量，文档转化为特征向量

–学习分类函数、

-在实际搜索系统中采用机器学习模型

它有以下3种方法：

（计算损失函数的方法，也是构造训练集的方法）

单文档方法

• PointWise Approach

• 损失函数评估单个 doc 的预测得分和真实得分之间差异

文档对方法

• PairWise Approach

• 是判断任意两个文档组成的文档对<D0C1，D0C2>是否满足顺序关系

文档列表方法

• ListWise Approach

• 搜索结果列表整体作为一个训练实例

第10讲 信息检索的评价

检索评测基础

、•信息检索系统的目标是较少消耗情况下尽快、全面返回准确的结果。

测试集由一个文档集、一组信息查询实例、对应于每个信息查询实例的\*\*一组相关文档（由专家提供）\*\*所组成

无序评测

查全率和查准率

•无序检索结果的评价

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-ri4IinkS-1608430839961)(image\image-20201217161456944.png)]

• 查准率(Precision):返回的结果中真正相关结果的比率，也称为查准率， P∈ [0,1]

• 召回率(Recall): 返回的相关结果数占实际相关结果总数的比率，也称为查全率，R∈ [0,1]

P = R R R R + R N R = R R R R + N R P=\frac{R R}{R R+R N} \quad R=\frac{R R}{R R+N R}

P=

RR+RN

RR

​

R=

RR+NR

RR

​

关于召回率的计算：增加一个缓冲池： •对多个检索系统的Top N个结果组成的集合进行人工标注，标注出的相关文档集合作为整个相关文档集合。查准率不变，召回率增大

精确率，不用它

平均

–宏平均(Macro Average): 对每个查询求出某个指标，然后对这些指标进行算术平均

–微平均(Micro Average): 将所有查询视为一个查询，将各种情况的文档总数求和，然后进行指标的计算

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-pBY2WnOS-1608430839962)(image\image-20201217162720957.png)]

F值(F-measure)

• F值(F-measure)：召回率R和查准率P的加权调和平均值，

• F1 标准则综合了精度和查全率，将两者赋予同样的重要性来考虑。F1的计算由下面的公式决定（调和平均数）

F ( i , j ) = 2 × recall ⁡ ( i , j ) × precision ( i , j ) recall ⁡ ( i , j ) + precision ⁡ ( i , j ) F(i, j)=\frac{2 \times \operatorname{recall}(i, j) \times \text { precision}(i, j)}{\operatorname{recall}(i, j)+\operatorname{precision}(i, j)}

F(i,j)=

recall(i,j)+precision(i,j)

2×recall(i,j)× precision(i,j)

​

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-8TG2e0UG-1608430839963)(image\image-20201217162932501.png)]

调和平均值

F = 2 1 r + 1 p F=\frac{2}{\frac{1}{r}+\frac{1}{p}}

F=

r

1

​

+

p

1

​

2

​

排序评测

R-查准率是什么

• 计算序列中第R个位置文献的查准率。在公式里指分母

• R是指与当前查询相关的文档总数.

• R=10, R-查准率=4/10;

• R=3, R-查准率=2/3

查准率/查全率曲线

横轴查全率，纵轴查准率

曲线下的面积被称为AP分数（Average precision score）

去掉锯齿，对一x取最大y

Mean Average Precision (MAP)是什么

• 平均查准率均值

• MAP是多个查询/排名的平均精度

• 在每个相关文档位置上查准率的平均值，被称为平均查准率 Average Precision (AP)

也就是对每个查询相关的R-查准率（在R位置上的那个文档是相关的）累计求和取均值

NDCG是什么

一种总体观察检索排序效果的方法，利用检索序列加和（每个搜索结果都要有个评价分，越高越好）的思路来衡量。

第11讲 概率检索模型

不考推导，只看思想，只有填空

看不懂，这点分，不要也罢

Probability ranking principle PRP概率排名原则

令x代表集合中的文档。令R代表文件w.r.t.的相关性。给定（固定）查询，令R = 1表示相关，而R = 0不相关。

• 概率检索模型作为一个分类问题，

• 对于某个文档d来说，如果其属于相关文档子集的概率大于属于不相关文档子集的概率，我们就可以认为这个文档与用户查询q

是相关的。

• P(R=1|q,d)代表给定一个文档D对应的相关性概率，

• P(R=0| q,d)则代表该文档的不相关概率

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-ZfmzRkaD-1608430839964)(image\image-20201216194643050.png)]

概率检索策略

估计每个词项对相关性的贡献

合并以查找文档相关性概率

通过概率降低顺序对文档进行排序

BIM Binary Independence Model 二元独立模型

Binary” =布尔值：文档表示为词项的二进制关联向量

Independence：term在文档中独立出现

词包模型

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-lpCcQel0-1608430839965)(image\image-20201216195435537.png)]

BM25

BM25是信息索引领域用来计算query与文档相似度得分的经典算法

不同于TF-IDF，BM25的公式主要由三个部分组成：

• query中每个单词t与文档d之间的相关性

• 单词t与query之间的相似性

• 每个单词的权重

目标：对术语频率和文档长度敏感，同时不添加太多参数

文件生成模型

​ 使用多项式分布从词典中独立绘制单词

​ 词项频率（tf）的分布遵循二项式分布-由泊\*\*松（Poisson）\*\*近似

泊松模型

​ 假设文档中的词频（tfi）遵循泊松分布

​ •“固定间隔”表示文档长度固定…认为大小恒定的文档摘要•…稍后将修复

第12讲 隐语义空间

奇异值分解需要了解，但是不考了

•用前r大的奇异值来近似描述矩阵

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-WX65Uzzn-1608430839966)(C:\Users\yandalao\AppData\Roaming\Typora\typora-user-images\image-20201220095654805.png)]

PCA主成分分析（回忆计算机视觉）

隐语义分析 LSA

什么是LSA

–使用统计计算的方法对大量的文本集进行分析，

–从而提取出词与词之间潜在的语义结构，并用这种潜在的语义结构，来表示词和文本，

达到消除词之间的相关性和简化文本向量实现降维的目的

把高维的向量空间模型（VSM）表示中的文档映射到低维的潜在语义空间中

基本步骤

（1）建立词频矩阵

（2）计算矩阵的奇异值分解

（3）对于每一个文档d，用排除了SVD中消除后的词的新的向量替换原有的向量

（4）用转换后的矩阵进行文档索引和相似度计算

LSA优点

（1）文档和单词都映射到同一个语义空间，所以可以计算文档和文档的相似度，词项和词项的相似度，词项和文档的相似度

（2）语义空间的维度明显明显少于源单词-文章矩阵

•最关键的性质：每个奇异值对应的是每个“语义”维度的权重

•将不太重要的权重置为0，可以保留重要的信息，去掉一些信息“枝节”。。枝节信息可能会使本来应该相似的对象不相似

LSA缺点

a) 无法解决多义词的问题

b) 特征向量的方向没有对应的物理解释

c) SVD的计算复杂度很高，而且当有新的文档来到时，若要更新模型需重新训练

d) 维数的选择是ad-hoc的

e) LSA具有词袋模型的缺点，即在一篇文章，或者一个句子中忽略词语的先后顺序

f) LSA的概率模型假设文档和词的分布是服从联合正态分布的，但从观测数据来看是服从泊松分布的

概率潜在语义分析 pLSA

什么是pLSA

a) PLSA是以统计学的角度来看待LSA，是基于双模式和共现的数据分析方法延伸的经典的统计学方法

生成模型

•在概率统计理论中，

–生成模型是指能够随机生成观测数据的模型，尤其是在给定某些隐含参数的条件下。

它给观测值和标注数据序列指定一个联合概率分布

什么是主题模型？

一篇文档(Document) 可以由多个主题(Topic) 混合而成

每个Topic 都是词汇上的概率分布

每个词都是由一个固定的 Topic 生成的

“文档-词项”的生成模型的训练？

a) 按照概率选择一篇文档d

b) 选定文档后，从主题分布中按照概率选择一个隐含的主题类别p(z|d)

c) 选定后，从词分布中按照概率p(w|z)选择一个词

PLSA生成文档的过程?

a) pLSA中生成文档的整个过程便是选定文档生成主题，确定主题生成词。

b) 自动地发现文档集中的主题（分布）

​ i. 根据大量已知的文档-词项信息p(w|d) ，

​ ii. 训练出文档-主题p(z|d)和主题-词项p(w|z)

EM算法

PLSA有哪些应用？

根据p(z|d)来的

a) 文本聚类

b) 文本分类

PLSA的优势？

a) 定义了概率模型，而且每个变量以及相应的概率分布和条件概率分布都有明确的物理解释

b) 相比于LSA隐含了高斯分布假设，pLSA隐含的Multi-nomial分布假设更符合文本特性

c) pLSA的优化目标是是KL-divergence最小，而不是依赖于最小均方误差等准则

d) 可以利用各种model selection和complexity control准则来确定topic

pLSA不足

•随着document和term 个数的增加，pLSA模型也线性增加，变得越来越庞大；

•PLSA可以生成其所在数据集的的文档的模型，但却不能生成新文档的模型。

•EM算法需要反复的迭代，需要很大计算量；

•概率模型不够完备

​ –不是完整的贝叶斯模型

​ –文档-主题p(z|d)和主题-词项p(w|z)是直接根据数据估计出来的，没有进一步引入先验

​ 这两点在LDA模型做了优化

LDA模型

什么是LDA模型？

a) 一个隐含狄利克雷分布的主题模型

和pLSA主题模型有什么区别

增加了狄利克雷的先验知识，所有的参数都不是设定的，而是进行了全贝叶斯化，更符合实际的情况

GENSIM

Gensim是一个用于从文档中自动提取语义主题的Python库

•第一步、准备训练语料

•第二步、预处理

​ –分词（tokenize the documents）、去除停用词和在语料中只出现一次的词

•第三步、文本向量化

第13讲 词嵌入

重点：统计语言，表征学习

统计语言模型

什么是语言模型和统计语言模型？

a) 语言模型根据语言客观事实而进行的语言抽象数学建模

b) 统计语言模型为上下文相关的特性建立数学模型

语言模型的公式

–S ：一连串特定顺序排列的词ω1，ω2，…，ωn

a) S 的概率 P(S)等于每一个词出现的概率相乘

b) P(S) =\*P\*(ω1)•\*P\*(ω2|ω1)•\*P\*(ω3|ω1,ω2)•••\*P\*(ωn|ω1，ω2，…，ωn-1)

什么是n-gram语言模型？

N-1阶马尔可夫假设:

​ 假定文本中的每个词ωi和前面的N-1个词有关，而与更前面的词无关

对应的语言模型称为N元模型(N-Gram Model)

统计语言模型、n-gram语言模型有什么应用

• 文本生成、机器翻译

• 拼写纠错

• 语音识别

• 音字转换

• 分词

n-gram语言模型的缺点

a) 简单有效

b) 只考虑了词的位置关系，

c) 没有考虑词之间的相似度，词语法和词语义，

d) 还存在数据稀疏的问题

文档重复检测

判断重复的思路：

–为每一个web文档通过hash的方式生成一个指纹（fingerprint）。

–将高维的特征向量映射成一个f-bit的指纹(fingerprint)，

通过比较两篇文章的f-bit指纹的Hamming Distance来确定文章是否重复或者高度近似

shingl算法

•核心思想是将文件相似性问题转换为集合的相似性问题

–给定正整数k及文档d的一个词项序列，可以定义文档d的k-shingle为d中所有k个连续词项构成的序列。

–a rose is a rose is a rose → 4-Grams

a\_rose\_is\_a

​ rose\_is\_a\_rose

​ is a rose is

​ a\_rose\_is\_a …

直观上看，如果两个文档的shingle集合几乎一样，那么它们就满足近似重复

局部敏感哈希 LSH

局部敏感哈希可以用来降维

MinHash的用处

a) 可以用来快速估算两个集合的相似度。

b) 用于在搜索引擎中检测重复网页。

c) 它也可以应用于大规模聚类问题

SimHash的步骤

a) 分词、hash、加权、合并、降维

w指的是每个term的权重

加权：遇到1则hash值和权值正相乘，遇到0则hash值和权值负相乘 例如W(CSDN) = 100101 4 = 4 -4 -4 4 -4 4

降维：对于n-bit签名的累加结果，如果大于0则置1，否则置0

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-IfucazqJ-1608430839967)(image\image-20201216220909219.png)]

相似度判断：每篇文档得到SimHash签名值后，接着计算两个签名的海明距离即可

表征学习和词嵌入

•表征学习：

–在机器学习中，表征学习是学习一个特征的技术的集合

–将原始数据转换成为能够被机器学习来有效开发的一种形式。

​ •向量

•嵌入（embedding）

​ –是一种可用于将离散变量表示成连续向量的方法。

神经网络语言模型

NNLM

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-7JBzTbHC-1608430839968)(image\image-20201217085938669.png)]

知道这个图各部分意思，下面的word2vec就是改进了一下上面

word2vec

•对原始的NNLM模型做如下改造：

–移除前向反馈神经网络中非线性的hidden layer（ tanh 隐藏层），直接将中间层的embedding layer与输出层的softmax layer连接；

–忽略上下文环境的序列信息：输入的所有词向量均汇总到同一个embedding layer；

–将future words纳入上下文环境

•连续词袋模型 CBOW

根据某个词前面的C个词或者前后C个连续的词，来计算某个词出现的概率

步骤，PPT非常清晰了

V是词项数量，N是中间向量那个O的维度

具体步骤：

模型输入：上下文的one hot表示方式

​ –1xV的向量

​ –V 词汇表大小

输入分别跟同一个VxN的大小的系数矩阵W1相乘得到C个1xN的隐藏层hidden layer，

然后C个取平均所以只算一个隐藏层

•隐藏层跟另一个NxV大小的系数矩阵W2相乘得到1xV的输出层，

​ –这个输出层每个元素代表的就是词库里每个词的事后概率。

•输出层需要跟ground truth也就是“coffee”的one hot形式做比较计算loss

•通过大量的数据迭代，使用梯度下降更新W和W’，来最小化loss函数，

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-Yf0THKo1-1608430839969)(image\image-20201217090553751.png)]

•Skip-Gram Model

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-8BKqtI1Y-1608430839970)(file:///D:\360MoveData\Users\yandalao\Documents\Tencent Files\2922610627\Image\C2C\AB502D3E6C82F00132C9127A669EA5E0.jpg)]

Skip-Gram Model相反，是根据某个词，然后分别计算它前后出现某几个词的各个概率

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-dR2lyz5a-1608430839970)(image\image-20201217091825010.png)]

Skip-gram–名称源于该模型在训练时会对上下文环境里的word进行采样

•基于成对的单词来对神经网络进行训练，

​ –训练样本是 ( input word, output word ) 这样的单词对，

​ –input word和output word都是one-hot编码的向量。

​ –最终模型的输出是一个概率分布。

​ •输出层使用了sotfmax。

•模型的本质:

​ 计算输入word和输出word的余弦相似度，并进行softmax归一化（想象一下softmax图像，所有的值都被分配到[0,1]之间的数）

•直接对词典里的 V 个词计算相似度并归一化，显然是一件极其耗时的impossible mission。为了加快速度优化：

负采样：

–层次Softmax（Hierarchical Softmax）

word2vec 应用

列出所有相似词语列表 和程序猿相似词语，比如攻城狮，比如猝死

•词汇的语义的类比 皇帝-皇后=男-女

•寻找对应关系： 男人——男孩 女人——女孩

第14讲 图片检索

图像检索

跨媒体检索Cross-Media Retrieval

–不同媒体映射到同一低维度空间

•基于文本的[图像检索技术]TBIR

​ –查询词：文本

​ –搜索引擎

​ •爬虫 图片

​ •索引 图片对应的文字，锚文本，URL

​ •基于图像周围文本的检索

​ •基于链接锚文本的检索

•基于内容的图像检索CBIR

–用户输入一张图片，以查找具有相同或相似内容的其他图片。

​ CBIR 的关键技术:图像特征提取和特征匹配

图像特征

•图像的特征主要包括低层特征(Primitive Features)和语义特征(Semantic Features）

–低层视觉

•与图像的具体类型或内容无关，

–颜色、形状、纹理等

•某些先验知识（或假设）

–人的面部特征

–指纹特征

图片的特征有颜色特征、形状特征、纹理特征

颜色特征

底层、直观，鲁棒性强

颜色特征的表示有几种

• 1、颜色直方图(Color Histogram) 直方图，就是CV教的那个，但是是对颜色来的，不是灰度

​ 没有体现空间信息，平移尺度旋转不变性

• \*\*2、颜色相关图(Color Correlogram)\*\*不考

• 3、颜色矩(Color Moment)

–在颜色直方图的基础上计算出每个颜色的矩估计

• 4、颜色一致性矢量(Color Coherence Vectors, CCV)

纹理特征

一般说纹理就是指在图像中反复出现的局部模式和它们的排列规则

基于统计特征的纹理特征提取

1.灰度差分统计法

[外链图片转存失败,源站可能有防盗链机制,建议将图片保存下来直接上传(img-DJPGNRYU-1608430839972)(image\image-20201217105234873.png)]

2.基于灰度共现矩阵的纹理特征 –常用统计量：对比度、相关度、方差、熵等

3.Tamura纹理特征

•Tamura纹理特征中所有纹理特征都在视觉上有意义。

–对比度(contrast)、粗糙度(coarseness)、方向性(directionality)对于图像检索尤为重要。

–线像度(1ine likeness)、规整度(regularity)和粗略度(roughness)。

基于信号处理方法描述纹理特征

–利用某种线性变换、滤波器或者滤波器组将纹理转换到变换域，

–然后应用某种能量准则提取纹理特征。

形状特征

有一定的语义信息

•基于轮廓的形状描述符

链码–差分结果第一位是原链码最后一位和第一位相减的结果。–例如，对于4向链码10030321的一阶差分的结果为03031333

基于网格的方法

傅里叶描述子

–物体轮廓线表示成一个一维的轮廓线函数

–傅立叶级数中的一系列系数z(k)是直接与边界曲线的形状有关的，称为傅立叶描述子．

•基于物体轮廓坐标序列的傅立叶描述子具有最佳的形状识别性能．

感知哈希算法

•全局特征降维

（1）对每张图片生成一个\*\*“指纹”（fingerprint）字符串，也就是图片的特征\*\*

（2）然后比较不同图片的指纹，结果越接近，就说明图片越相似（用海明距离来计算）

（之前计算文档相似度的局部敏感哈希也是用hash法，比较哈希码的相似度来判断文档相似程度，都是用海明距离）

那么怎么将图片变为哈希码呢？

（1）均值Hash算法

缩小尺寸，收缩色彩度（比如300-64），计算所有像素的灰度平均值，阈值二值化，二值化结果为哈希值

（2）pHash算法

（3）颜色分布法–红绿蓝分别有4个区（颜色分段）

–总共可以构成64种组 4^3。

•任何一种颜色必然属于这64种组合中的一种——特征为64维向量，计算余弦相相似度

​ (4)•内容特征法

(图片二值化)–原图转成一张较小的灰度图片，确定一个阈值，将灰度图片转成黑白图片

–两张图片很相似，它们的黑白轮廓应该是相近的

•基于区域的形状描述符

大津法Otsu’s method

a) 证明了 "类内差异最小"与"类间差异最大"是同一件事

b) 计算方法：

​ i. 灰度值小于阈值的像素为 n1 个，

​ ii. 大于等于阈值的像素为 n2 个

​ iii. w1 和 w2 表示这两种像素各自的比重

​ iv. w1 = n1 / n

​ v. 类内差异 = w1(σ1的平方) + w2(σ2的平方)

​ vi. 类间差异 = w1w2(μ1-μ2)^2

图像局部特征

LBP特征

局部二值模式 Local Binary Patterns，结合了纹理图像结构和像素统计关系的纹理特征描述方法

LBP怎么构造

• LBP算子定义为在3\*3的窗口内，

• 以窗口中心像素为阈值，将相邻的8个像素的灰度值与其进行比较，若周围像素值大于中心像素值，则该像素 点的位置被标记为1，否则为0。

• 3\*3邻域内的8个点经比较可产生8位二进制数（通常转换为十进制数即LBP码，共256种），即得到该窗口中心像 素点的LBP值，并用这个值来反映该区域的纹理信息。

LBP的应用中，如纹理分类、人脸分析等，采用LBP特征谱的统计直方图作为特征向量用于分类识别。可将一幅图片化为多个子区域，分别求每个子区域的统计直方图。

HOG特征

关键词：cell，梯度直方图，行人检测

HOG是什么？

a) 方向梯度直方图，Histogram of Oriented Gradient, HOG

b) 一种在计算机视觉和图像处理中用来进行物体检测的特征描述子

c) 通过计算和统计图像局部区域的梯度方向直方图来构成特征

Hog特征结合 SVM分类器已经被广泛应用于图像识别中，尤其在行人检测中获得了极大的成功

HOG特征如何提取？

a) 灰度化

b) 采用Gamma校正法对输入图像进行颜色空间的标准化（归一化）

c) 计算图像每个像素的梯度

d) 将图像划分成小cells

e) 统计每个cell的梯度直方图

梯度直方图，横轴是梯度方向，y轴是在该梯度方向的梯度值的和

f) 将每几个cell组成一个block

g) 将图像image内的所有block的HOG特征descriptor串联起来就可以得到该image的HOG特征descriptor了

HOG算法的优缺点？

a) 优点

​ i. 由于HOG是在图像的局部方格单元上操作，所以它对图像几何的和光学的形变都能保持很好的不 变性，这两种形变只会出现在更大的空间领域上。

​ ii. 其次，在粗的空域抽样、精细的方向抽样以及较强的局部光学归一化等条件下，只要行人大体上能够保持直立的姿 势，可以容许行人有一些细微的肢体动作，这些细微的动作可以被忽略而不影响检测效果。

​ iii. 因此HOG特征是特别适合于做图像中的人体检测的

SIFT

SIFT特征是什么

尺度不变特征转换，Scale-invariant feature transform或SIFT，在空间尺度中寻找极值点，并提取出其位置、尺度、旋转不变量。

SIFT特征和HOG特征好处

SIFT特征不只具有尺度不变性，即使改变旋转角度，图像亮度或拍摄视角，仍然能够得到好的检测效果，Hog没有旋转和尺度不变性

SIFT有哪几个步骤

– 步骤一：建立尺度空间

• 即建立高斯差分(DoG)金字塔

– 步骤二：在尺度空间中检测极值点，并进行精确定位和筛选

– 步骤三：特征点方向赋值，

• 完成此步骤后，每个特征点有三个信息：位置、尺度、方向

– 步骤四：计算特征描述子

SIFT特征的匹配是暴力匹配

图像检索算法

图像检索算法

a) 图像检索领域：将局部特征表示成全局特征的编码

b) 通常继承了局部特征的部分不变性，如对平移、旋转、缩放、光照和遮挡等与语义相关不大的因素保持不变

三种经典的编码

a) [BoW](http://yongyuan.name/blog/Bag of visual words model: recognizing object categories)

b) VLAD局部聚合向量

c) FV

BOF

图像视为文档，局部特征经过聚类后看作一个视觉词汇（也就是词）

BOF算法先求出特征点，再聚类生成类心，得到视觉词汇，生成直方图（横轴视觉词汇，纵轴频数），再根据TF-IDF调整权重

查询时，求夹角余弦

BOF算法流程

– 1.用surf算法生成图像库中每幅图的特征点及描述符。

• surf算法是关键点计算和描述算法，作用和SIFT相似。

– 2.再用k-means算法对图像库中的特征点进行训练，生成类心。

– 3.生成每幅图像的BOF，

• 判断图像的每个特征点与哪个类心最近，最近则放入该类心，最后将生成一列频数表，即初步的无权BOF（直方图向量）。

– 4.通过tf-idf对频数表加上权重，生成最终的bof。

• 因为每个类心对图像的影响不同。比如超市里条形码中的第一位总是6，它对辨别产品毫无作用，因此权重要减小。

• TF/IDF

– 5.对查询图像也进行3.4步操作，生成该图的直方图向量BOF。

– 6.将查询图像的Bof向量与图像库中每幅图的Bof向量计算相似度

• 求夹角余弦。

Fisher vector

FV考虑了特征点到每个聚类中心的距离，也就是用所有聚类中心的线性组合去表示该特征点

​ –FV描述局部特征和GMM中心之间的平均一阶和二阶差异

VLAD特征

•可以认为VLAD是FV的简化版本

•如同BOF先建立出含有k个visual word的codebook，只考虑离特征点最近的聚类中心

​ -采用的是计算出local descriptor和每个visual word(ci)在每个分量上的差距，将每个分量的差距形成一个新的向量来代表图片

————————————————

版权声明：本文为CSDN博主「sssssssssad」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：https://blog.csdn.net/weixin\_43756517/article/details/111319904