黑客派

Lucene 中的评分模型—Vector Space Model(空间向量模型)

作者：felayman

原文链接：https://hacpai.com/article/1519795871451

来源网站：黑客派

<h2 id="基本思想">基本思想</h2>
<blockquote>
在自然界中任何事物都可以用一些最基本的元素加以表示，这些最基本的元素作为基础单元，似于坐标系中坐标轴，通过这种假设与推理，每一个构成事物的基本元素都对应着 n 维空间中某个坐系，则事物可通过各个基本元素表示为坐标系向量的形式。
</blockquote>
那么，两个向量之间的夹角越小，则两个向量所代表的事物就越相似
在 Lucene 中，这个思想可以简化如下：
把文档看成是一个向量（vector），其中的每个分量都对应词典中的一个词项，分量值为采用 tf-df 计算出的权重值。当某词项在文档中没有出现时，其对应的分量值为 0。
于是，我们有一个 |V| 维实值空间，空间的每一维都对应词项（V 为词项数目）。
对于 Web 搜索引擎，空间可能会上千万维。但对每个向量来说又非常稀疏（稀疏矩阵），大部都是 0。
<h2 id="基本概念">基本概念</h2>

词向量

<blockquote>
这里的词向量指的是最初是在检索系统中为了计算 query 和文档的相关性的概念，关于在自然言中的概念，可以参考词向量（Distribute Representation）工作原理是什么？
</blockquote>

文本

<blockquote>
通常是文本中具有一定规模的片断，如句子、句群、段落段落组直至整篇文本
</blockquote>

项/特征项

<blockquote>
特征项是文本表示中最基本的元素，正是由于特征项之间的不同组合构成了文本，同时特征项为基本元素构成了表示文本的向量形式。 文本被看作为项的集合 Document = (t1,t2,t3...tn)
</blockquote>

项的权重

<blockquote>
Document = (t1,t2,t3...tn)表示文档中包含 n 个关键词（特征项），在文本向量中每一个维度的特征项 tk 都依据一定的原则被赋予一个特征项权重 wk 表示它们在文档中的重要程度。权值的计方法有几种：基于词频（TF）的关键词权值，基于文档频率（DF）的关键词权值，基于文档频率的键词权值，基于信息增益的关键词权值，基于卡方分布的关键词权值，基于互信息的关键词权值
</blockquote>
我们可以（t1,t2,t3..tn）看成是一个 n 维坐标系。坐标系的每一个维度对应一个特征项，权重对在坐标轴上的值。 一个文本就是坐标系中的一个向量。
D = (w1,w2,w3..wn)就是文本的向量表示
<h2 id="如何计算相似度">如何计算相似度</h2>
设文档 D1 和 D2 表示向量空间模型中的两个向量
D1 = (w11,w12,w13..w1n)
D2 = (w21,w22,w23..w2n)
那么两个文本的相似度计算公式如下：

oad-images.jianshu.io/upload_images/1354300-e645d6c9b56c9dc9.PNG?imageMogr2/auto-rient/strip%7CimageView2/2/w/443">
那么现在需要了解的就是如何计算出文档中的每个词向量的值，Lucene 这里是是有 TF-IDF 模来，详细内容可以参考：lucene4.5 源码分析系列：lucene 的默认评分算法-向量空间模型（Vector Space Model）
向量空间模型
现在的问题就变成，如何求得每个维度上的 term 在文档中的权重，在向量空间模型中，特征权的计算框架是 TF*IDF 框架，这里 TF 就是 term 在文档中的词频，TF 值越大，说明该篇文档相对于个 term 来说更加重要，因此，权重应该更高；而 IDF 则是 term 在整个文档集中占的比重，即 n/N 其中 n 是含该 term 的文档数，N 是总文档数，但是，实际使用中往往习惯用
<img src="https://static.hacpai.com/images/img-loading.svg" alt="" data-src="http://im.blog.csdn.net/20130930163441312?watermark/2/text/aHR0cDovL2Jsb2cuY3Nkbi5uZXQvbGlZWlzbmFrZQ==/font/5a6L5L2T/fontsize/400/fill/I0JBQkFCMA==/dissolve/70/gravity/SouthEst">
即所包含的该 term 的文档数越少说明该 term 越重要。可以举个例子，有 100 篇文档，其中 80 篇都在说红楼梦，其中只有几篇讲到计算机，当你在这个文档集中搜索到计算机时，可以肯定这几篇计算机的比较重要，而搜索红楼梦时，则很难区分哪篇更加重要，换句话说，在这个文档集合中，计机比红楼梦更有区分度，相对来说，计算机比红楼梦更有信息量，所以 IDF 就是评判所含信息量大小一个值。
一般情况，使用 TF*IDF 作为这里的权重 w，从而计算出 dj,q 的相似度 sim(dj,q)。
那么，在 lucene 中，是如何应用这个模型的呢？根据向量空间模型的的数学推导（见参考文档），可以看到，在 lucene 中实际上是将 sim(dj,q)变形和调整后应用了如下一个打分公式
<img src="https://static.hacpai.com/images/img-loading.svg" alt="" data-src="http://im.blog.csdn.net/20131009155338046?watermark/2/text/aHR0cDovL2Jsb2cuY3Nkbi5uZXQvbGlZWlzbmFrZQ==/font/5a6L5L2T/fontsize/400/fill/I0JBQkFCMA==/dissolve/70/gravity/SouthEst">
关于该公式的详细介绍，可以参考我的另外一篇文章： lucene 的评分机制
注意：
Lucene 从 7.0.0 之后的 TFIDF 评分公式发生了变化，比如 Lucene7.0.0 版本以前的评分公式为
<img src="https://static.hacpai.com/images/img-loading.svg" alt="" data-src="http://felyman.oss-cn-beijing.aliyuncs.com/Lucene/2.png">
Lucene7.0.0 之后的版本为：
<img src="https://static.hacpai.com/images/img-loading.svg" alt="" data-src="http://felyman.oss-cn-beijing.aliyuncs.com/Lucene/1.png">
可以看到，Lucene7.0.0 移除了 coord(q,d) · queryNorm(q)，详细变化的原因官方有给出，请考：LUCENE-7347 Remove queryNorm and coordsLUCENE-7369
 Remove coordination factors from scores
<h2 id="总结">总结</h2>

将查询表示成 tf-idf 权重向量
将每篇文档表示成同一空间下的 tf-idf 权重向量
计算两个向量之间的某种相似度(如余弦相似度)
按照相似度大小将文档排序
将前 K（如 K =10）篇文档返回给用户

<h2 id="参考">参考</h2>

 <a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fen.wikipedia.org2Fwiki%2FVector_space_model" target="_blank" rel="nofollow ugc">Vector space model</a
 文本分类三之向量间模型
 向量空模型（VSM）算法
 向量空间模型-我爱自语言处理
 向量空间型
 漫谈词向量
 Deep Learning in NLP （一）词向量和语言型
 <a href="https://link.hacpai.com/forward?goto=https%3A%2F%2Fcloud.baidu.co%2Fproduct%2Fnlp%2Fword_embedding" target="_blank" rel="nofollow ugc">词向量表示</>
 词向量（Distributed Representation）工作原理是什么？
 基于向量空间模型的余弦相似度法
 如何通过词向量技术来计算 2 文档的相似度?
