京东云成都团队的专栏

个人资料

京东云成都

访问： 23297次
积分： 308
等级： BLOG 2
排名： 千里之外

原创： 7篇
转载： 0篇
译文： 0篇
评论： 7条

余弦计算相似度度量

标签： 算法

2013-11-11 19:10 7550人阅读 评论(3) 收藏 举报

目录(?) [+]

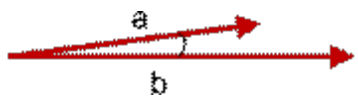
余弦计算相似度度量

相似度度量（Similarity），即计算个体间的相似程度，相似度度量的值越小，说明个体间相似度越小，相似度的值越大说明个体差异越大。

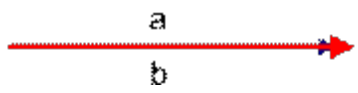
对于多个不同的文本或者短文本对话消息要来计算他们之间的相似度如何，一个好的做法就是将这些文本中词语，映射到向量空间，形成文本中文字和向量数据的映射关系，通过计算几个或者多个不同的向量的差异的大小，来计算文本的相似度。下面介绍一个详细成熟的向量空间余弦相似度方法计算相似度

向量空间余弦相似度**(Cosine Similarity)**

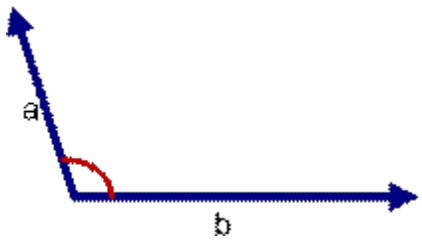
余弦相似度用向量空间中两个向量夹角的余弦值作为衡量两个个体间差异的大小。余弦值越接近1，就表明夹角越接近0度，也就是两个向量越相似，这就叫"余弦相似性"。



上图两个向量a,b的夹角很小可以说a向量和b向量有很高的的相似性，极端情况下，a和b向量完全重合。如下图：



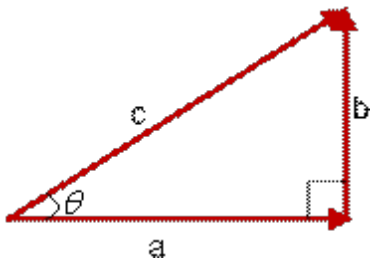
如上图二：可以认为a和b向量是相等的，也即a，b向量代表的文本是完全相似的，或者说是相等的。如果a和b向量夹角较大，或者反方向。如下图



如上图三: 两个向量a,b的夹角很大可以说a向量和b向量有很底的的相似性，或者说a和b向量代表的文本基本不相似。那么是否可以用两个向量的夹角大小的函数值来计算个体的相似度呢？

向量空间余弦相似度理论就是基于上述来计算个体相似度的一种方法。下面做详细的推理过程分析。

想到余弦公式，最基本计算方法就是初中的最简单的计算公式，计算夹角 θ



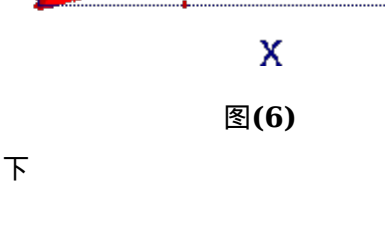
图**(4)**

的余弦定值公式为：

$$\cos(\text{最小化}) = \frac{a}{c}$$

公式(1)

但是这个是只适用于直角三角形的,而在非直角三角形中,余弦定理的公式是



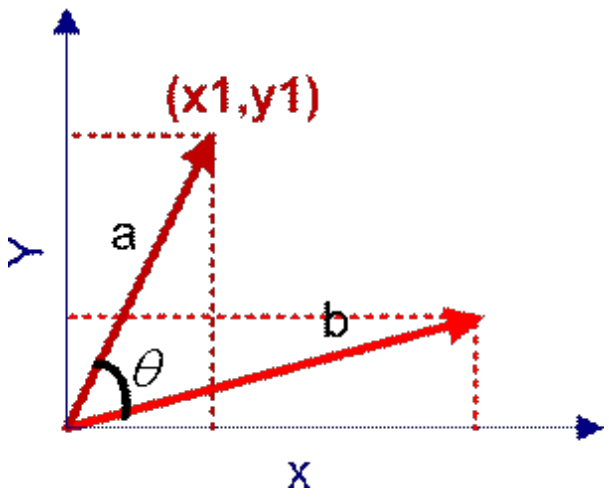
图(5)

三角形中边a和b的夹角 的余弦计算公式为：

$$\cos(\theta) = \frac{a^2 + b^2 - c^2}{2ab}$$

公式(2)

在向量表示的三角形中，假设a向量是（x1, y1），b向量是(x2, y2)，那么可以将余弦定理改写成下



图(6)

向量a和向量b的夹角 的余弦计算如下

$$\cos(\theta) = \frac{a \bullet b}{||a|| \times ||b||}$$

最小化

$$= \frac{(x_1, y_1) \bullet (x_2, y_2)}{\sqrt{x_1^2 + y_1^2} \times \sqrt{x_2^2 + y_2^2}}$$

$$= \frac{x_1 x_2 + y_1 y_2}{\sqrt{x_1^2 + y_1^2} \times \sqrt{x_2^2 + y_2^2}}$$

公式(3)

扩展，如果向量a和b不是二维而是n维，上述余弦的计算法仍然正确。假定a和b是两个n维向量，a是 ，b是 ，则a与b的夹角 的余弦等于：

最小化

$$\cos(\theta) = \frac{\sum_{i=1}^{n}(X_i \times Y_i)}{\sqrt{\sum_{i=1}^{n}(X_i)^2} \times \sqrt{\sum_{i=1}^{n}(Y_i)^2}}$$

$$= \frac{a \bullet b}{||a|| \times ||b||}$$

公式(4)

余弦值越接近1，就表明夹角越接近0度，也就是两个向量越相似，夹角等于0，即两个向量相等，这就叫"余弦相似性"。

【下面举一个例子，来说明余弦计算文本相似度】

举一个例子来说明，用上述理论计算文本的相似性。为了简单起见，先从句子着手。

句子**A**：这只皮靴号码大了。那只号码合适

句子**B**：这只皮靴号码不小，那只更合适

怎样计算上面两句话的相似程度？

基本思路是：如果这两句话的用词越相似，它们的内容就应该越相似。因此，可以从词频入手，计算它们的相似程度。

第一步，分词。

句子**A**：这只**/**皮靴**/**号码**/**大了。那只**/**号码**/**合适。

句子**B**：这只**/**皮靴**/**号码**/**不**/**小，那只**/**更**/**合适。

第二步，列出所有的词。

这只，皮靴，号码，大了。那只，合适，不，小，很

第三步，计算词频。

句子**A**：这只**1**，皮靴**1**，号码**2**，大了**1**。那只**1**，合适**1**，不**0**，小**0**，更**0**

句子**B**：这只**1**，皮靴**1**，号码**1**，大了**0**。那只**1**，合适**1**，不**1**，小**1**，更**1**

第四步，写出词频向量。

句子**A**：**(1，1，2，1，1，1，0，0，0)**

句子**B**：**(1，1，1，0，1，1，1，1，1)**

到这里，问题就变成了如何计算这两个向量的相似程度。我们可以把它们想象成空间中的两条线段，都是从原点（[0, 0, ...]）出发，指向不同的方向。两条线段之间形成一个夹角，如果夹角为0度，意味着方向相同、线段重合,这是表示两个向量代表的文本完全相等；如果夹角为90度，意味着形成直角，方向完全不相似；如果夹角为180度，意味着方向正好相反。因此，我们可以通过夹角的大小，来判断向量的相似程度。夹角越小，就代表越相似。

使用上面的公式**(4)**

$$\cos(\theta) = \frac{\sum_{i=1}^{n}(X_i \times Y_i)}{\sqrt{\sum_{i=1}^{n}(X_i)^2} \times \sqrt{\sum_{i=1}^{n}(Y_i)^2}}$$

计算两个句子向量

句子**A**：**(1，1，2，1，1，1，0，0，0)**

和句子**B**：**(1，1，1，0，1，1，1，1，1)**的向量余弦值来确定两个句子的相似度。

计算过程如下：

$$\cos(\theta) = \frac{1\times1+1\times1+2\times1+1\times0+1\times1+1\times1+0\times1+0\times1+0\times1}{\sqrt{1^2+1^2+2^2+1^2+1^2+1^2+0^2+0^2+0^2} \times \sqrt{1^2+1^2+1^2+0^2+1^2+1^2+1^2+1^2+1^2}}$$

$$= \frac{6}{\sqrt{7} \times \sqrt{8}}$$

$$= 0.81$$

关闭

计算结果中夹角的余弦值为0.81非常接近于1，所以，上面的句子A和句子B是基本相似的

由此，我们就得到了文本相似度计算的处理流程是:

（1）找出两篇文章的关键词；

（2）每篇文章各取出若干个关键词，合并成一个集合，计算每篇文章对于这个集合中的词的词频

（3）生成两篇文章各自的词频向量；

（4）计算两个向量的余弦相似度，值越大就表示越相似。

上一篇 MapR设计分析

下一篇 谈谈A/B Test

相关文章推荐

- 网页去重（四）之余弦夹角计算相似度
- 余弦方法计算相似度算法实现
- Java实现余弦定理计算文本相似度
- 余弦计算相似度度量
- 余弦计算相似度度量(优秀)
- 余弦计算相似度度量【转】
- 余弦计算相似度度量
- Python简单实现基于VSM的余弦相似度计算
- JAVA计算稀疏矩阵余弦相似度
- 基于向量余弦的文件相似度计算

猜你在找

数据结构与算法在实战项目中的应用

使用决策树算法对测试数据进行分类实战

使用决策树算法对测试数据进行分类实战

数据结构和算法

数据结构基础系列(1)：数据结构和算法

以性别预测为例，谈谈数据挖掘中常见的分类算法

Python算法实战视频课程--二叉树

使用 FP-growth 算法进行频繁项集挖掘实战

使用 AdaBoost 算法进行二分类实战

使用逻辑回归算法进行融资成功概率分析实战

查看评论

3楼 chao_beyond 2014-05-27 00:10发表

应该是根号9不是根号7吧

2楼 sunhongtt 2014-03-23 21:37发表

如果是：

A：我很想喝水

B：我有点渴
汇总 我 很 想 喝水 有点 渴
A 1 1 1 1 0 0
B 1 0 0 0 1 1
结果是 0.4多啊。这如何解决。
文章中的例子很好！通俗易懂

1楼 guwenwu285 2013-11-18 09:58发表

余弦相似度计算貌似在全文索引里面用的很多啊

发表评论

用 户 名： haijunz

评论内容：

提交

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

全部主题 Hadoop AWS 移动游戏 Java Android iOS Swift 智能硬件 Docker OpenStack
VPN Spark ERP IE10 Eclipse CRM JavaScript 数据库 Ubuntu NFC WAP jQuery
BI HTML5 Spring Apache .NET API HTML SDK IIS Fedora XML LBS Unity
Splashtop UML components Windows Mobile Rails QEMU KDE Cassandra CloudStack
FTC coremail OPhone CouchBase 云计算 iOS6 Rackspace Web App SpringSide Maemo
Compuware 大数据 aptech Perl Tornado Ruby Hibernate ThinkPHP HBase Pure Solr
Angular Cloud Foundry Redis Scala Django Bootstrap

关闭