

## BM25

首先，让我们来看一下BM25的公式，随后我将会细细分解这个公式：

$$\sum_i^n IDF(q_i) \frac{f(q_i, D) * (k1 + 1)}{f(q_i, D) + k1 * (1 - b + b * \frac{fieldLen}{avgFieldLen})}$$

- $q_i$  表示第  $i$  个query term。

比如搜索"Hogwarts School"，ElasticSearch会按照white space将其划分，于是得到两个token："Hogwarts"，"School"。由公式可知，BM25值就是将所有token的得分计算加和。

- $IDF(q_i)$  是第  $i$  个query term的逆文档频率(inverse document frequency)。

这里的IDF和TF-IDF中的IDF类似，都是用来惩罚那些出现在很多document中的词语，只是有一些小小的不同。Lucene/BM25的IDF计算公式如下：

$$\ln \left( 1 + \frac{(docCount - f(q_i) + 0.5)}{f(q_i) + 0.5} \right)$$

其中， $docCount$  是在ElasticSearch的一个shard (或者多个shards) 中的document个数； $f(q_i)$  是含有  $q_i$  的document的个数。

举个例子，假如总共有4个document，"school"出现在2个document中，那么IDF("school")为：

$$\ln \left( 1 + \frac{(4 - 2 + 0.5)}{2 + 0.5} \right) = \ln \left( 1 + \frac{2.5}{2.5} \right) = 0.693147180559945$$

也就是说，我们要给罕见的term分配较高的权重。

- $fieldLen/avgFieldLen$

在分母中的  $fieldLen/avgFieldLen$  其实是【给那些长document以惩罚】（这里的length是用term个数衡量的）。这也是符合我们的直觉的：假如一篇300页的文章提过一次query中的词，那肯定不如一个短短的句子提过query更相关。

- $b$ : 这是一个决定  $fieldLen/avgFieldLen$  影响大小的超参数。b越大，document长度的惩罚就越大。在ElasticSearch中，b的default值取0.75。
- $f(q_i, D)$

第

$i$

个 query term在document D中出现的次数。当然越多越好。

- $k_1$ : 用来决定[term frequency saturation](#)。即，限制了一个query term最多能够对最后的score有多大的影响。例如，一个文章中出现了20次query term和出现1000次query term的效果应该是差不多的。如果不做此限制，那么那些高频的词的tf值就会过大，导致整个query的得分都被那些高频词所主导。ElasticSearch中，default  $k_1 = 1.4$ 。BM25和TFIDF的对比如下图所示：

