1. **相关度评分算法揭秘**
   1. **boolean model: 所谓bool 就是 true false**
      1. **match operator: or and not**
      2. **bool => must/must not/should**
      3. **term->doc分数**
   2. **boost：权重**
   3. **TF-IDF**
      1. TF(词频term frequency): 关键词在每个doc中出现的次数
      2. IDF(反文档词频inversed document frequency):关键词在整个索引中出现的次数
      3. norm:字段长度越长，值越小。
      4. 空间向量模型 或者向量空间模型
         1. 比如 query: hello word
      5. 计算公式:

score(q,d) =

queryNorm(q)

· coord(q,d)

· ∑ (

tf(t in d)

· idf(t)²

· t.getBoost()

· norm(t,d)

) (t in q)

* + - 1. score(q,d)：query对一个doc最终的评分结果。
      2. queryNorm(q)：想想normalization，在不影响相互关系的前提下，把看似离散的数据，转换到一个相近的区间=>人性化
         1. queryNorm = 1 /√sumOfSquaredWeights
         2. sumOfSquaredWeights是通过将查询中每个项的IDF平方相加来计算的。
      3. coord(q,d)： 对匹配的结果加分，越匹配的doc加分越多

doc with hello→ score: 1.5 \* 1 / 3 = 0.5

doc with hello word → score: 3.0 \* 2 / 3 = 2.0

doc with hello word elastic → score: 4.5 \* 3 / 3 = 4.5

总分数 \* 匹配的term数 / 总term数

* + - 1. ∑：doc对query中每个trem的权重的总和
      2. tf(t in d)：
         1. tf(t in d) = √frequency
         2. 该trem在doc中出现的次数的平方根
      3. idf(t)：
         1. idf(t) = 1 + log ( numDocs / (docFreq + 1))
         2. trem的反文档频率(idf)是索引中文档数量的对数，除以包含该trem的doc数量
      4. t.getBoost()：设置的权重值.
      5. norm(t,d)：字段长度越长，结果月越小
         1. norm(d) = 1 / √numTerms
         2. 字段长度范数(范数)是字段中项数的平方根的倒数

1. Java API

##### Transport Client：TransportClient不推荐使用，而推荐使用[Java High Level REST Client](https://www.elastic.co/guide/en/elasticsearch/client/java-rest/7.6/java-rest-high.html)，并将在Elasticsearch 8.0中删除。

* 1. **JAVA REST Client**
     1. **Java Low Level REST Client**：低级别的REST客户端，通过http与集群交互，用户需自己编组请求JSON串，及解析响应JSON串。**兼容所有ES版本**
     2. **Java High Level REST Client：**高级别的REST客户端，基于低级别的REST客户端，增加了编组请求JSON串、解析响应JSON串等相关api。**使用的版本需要保持和ES服务端的版本一致，否则会有版本问题。**
        1. **要求最低JDK 1.8**
        2. **Maven配置**

**<dependency>**

**<groupId>org.elasticsearch.client</groupId>**

**<artifactId>elasticsearch-rest-high-level-client</artifactId>**

**<version>7.6.2</version>**

**</dependency>**