

# 搜索排序指标MRR,MAP,NDCG,RC的个人理解

风清云 风清云的学习之路 2019-04-24

入坑推荐系统以后，一直没有对搜索排序这一块有系统的理解，今天下午才开始阅读微软亚研院刘铁岩的《Learning to Rank for Information Retrieval》，准备对搜索排序的原理，算法等做一个初步且较为全面的了解。

讲到排序结果，必然需要有指标用于评价该排序结果的好坏。书中提到常用的评价指标主要有 Mean Reciprocal Rank(MRR)、Mean Average Precision(MAP)、Normalized Discounted Cumulative Gain(NDCG)以及Rank Correlation(RC)。

假设现在有这么一个场景，用户使用搜索引擎发起一个query（比如百度搜索“复仇者联盟四”），模型针对这个query找出了一系列（可能相关）的documents（即搜索出来的结果是按顺序排列的），并且按顺序把这些documents推送给用户，按顺序排列每个document会有它的真实得分（即与query的真实相关程度）如下表所示：

模型推送顺序	该文档实际得分
1	4.0
2	3.0
3	2.0
4	1.0
5	3.0
6	1.0
7	2.0

即模型推送在前七个的documents得分依次为 4.0,3.0,2.0,...,1.0。假设当真实得分大于2分时，认为该文档与用户的query相关（记为1），否则不相关（记为0），上面的表格可以变为如下形式：

模型推送顺序	该文档相关与否
1	1
2	1
3	0
4	0
5	1
6	0
7	0

有了数据和标签以后，就可以依次计算评价指标了：

#### 1, MRR

这是所有指标中最简单的一个，找出该query相关性最强的文档所在位置，并对其取倒数，即这个query的MRR值。本例中，真实得分最高的文档为4分，且排在第1位，那么这个query的MRR值即  $1 / 1 = 1$ ，如果排在第*i*位，则  $MRR = 1 / i$ 。然后对所有query的MRR值取平均即可得到该数据集上的MRR指标，显然MRR越接近1模型效果越好。

#### 2, MAP

这个指标需要通过几个公式定义来解释。首先为了定义MAP需要确定一个参数*k*，*k*代表我们只关注前*k*个documents中的MAP值。接下来定义*P@k*：

$$P@k(\pi, l) = \frac{\sum_{t \leq k} I_{\{l_{\pi^{-1}(t)} = 1\}}}{k},$$

这里 $\pi$ 代表documents list，即推送结果列。 $I_{\{ \}}$ 是指示函数， $\pi^{-1}(j)$ 代表位置*j*处的document的标签（相关为1，否则为0）。

再定义Average Precision(AP)：

$$AP(\pi, l) = \frac{\sum_{k=1}^m P@k \cdot I_{\{l_{\pi^{-1}(k)}=1\}}}{m_1},$$

风清云的硕士生涯

其中 $m_1$ 代表与该query相关的document的数量（即真实标签为1）， $m$ 则代表与query有联系的所有document的数量，本例中 $m=7$ ，并假设 $m_1=6$ ，即真正和query相关的document仅有6个。

那么我们可以计算该query的AP值为：

$$\begin{aligned} & \frac{1}{6} \left( P@1 \cdot I_{\{l_{\pi^{-1}(1)}=1\}} + P@2 \cdot I_{\{l_{\pi^{-1}(2)}=1\}} + \dots + P@7 \cdot I_{\{l_{\pi^{-1}(7)}=1\}} \right) \\ &= \frac{1}{6} \left( \frac{1}{1} \cdot 1 + \frac{2}{2} \cdot 1 + \frac{2}{3} \cdot 0 + \dots + \frac{3}{7} \cdot 0 \right) \\ &= \frac{13}{30} \end{aligned}$$

风清云的硕士生涯

得出该query的AP值以后，MAP值就是把所有query的AP值都计算出来再取平均即可。

再来看看网上给出的例子，假设query1有4个相关的document，分别被模型排在1, 2, 5, 7位，那么query1的AP就是 $(1/1+2/2+3/5+4/7)/4$ ，query2有5个相关的document，分别被排在2, 3, 6, 29, 58位那么query2的MAP就是 $(1/2+2/3+3/6+4/29+5/58)/5$ ，但通常情况下，我们的 $m$ 不会取到58，只会关注排名靠前的document，因此排在29与58的document可以视为没有被模型检索出来，假设取 $m=8$ ，则query2的AP是 $(1/2+2/3+3/6+0+0)/5$ 。对以上两个query取平均即可得出MAP。

### 3, NDCG

这个指标可以参考链接：

<https://www.cnblogs.com/by-dream/p/9403984.html>

### 4, RC

这个指标网上似乎没有相关资料详细讲解，估计是使用的频率比较少，看明白这个花了不少功夫，首先定义：


$$\tau_K(\pi, \pi_l) = \frac{\sum_{u < v} w_{u,v} (1 + \text{sgn}((\pi(u) - \pi(v))(\pi_l(u) - \pi_l(v))))}{2 \sum_{u < v} w_{u,v}},$$

风清云的硕士生涯

其中 $\pi$ 代表模型得出的document list， $\pi_l$ 代表某个最优的document list排序结果（因为当文档相关性取0-1的时候，有多个最优排序，比如上述例子中最优排序可以是1, 2, 5, 3, 4, 6, 7或者5, 2, 1, 7, 4, 6, 3）。 $\pi(u)$ 表示document  $u$ 在模型得出

的list中的位置，这里我们假设document的下标和模型给出的排序顺序是一样的，即排第1的为document 1。

这次换一个简单的例子来讲，因为使用上述例子实在难以计算QAQ，场景如下：

模型推送顺序	该文档相关与否
1	1
2	0
3	 风清云的硕士生涯

那么模型的推送文档顺序为(doc1, doc2, doc3) (简写为 (1, 2, 3) )，而最优的文档排序应该为(1,3,2)或(3,1,2)。所以 $\pi=(1,2,3)$ ， $\pi_l$ 有两个(1,3,2)和(3,1,2)。RC的值就定义为

$$\max_l \left( \tau_K \left( \pi, \pi_l \right) \right)$$

即把 $\pi$ 分别和 $\pi_l$ 计算后取最大值即可。以(1,3,2)为例计算：

$$\begin{aligned} \tau_K(\pi, \pi_l) &= \frac{w_{1,2} \left( 1 + \text{sgn}[(\pi(1) - \pi(2)) \cdot (\pi_l(1) - \pi_l(2))] \right) + w_{2,3}(\dots) + w_{1,3}(\dots)}{2 \cdot (w_{1,2} + w_{2,3} + w_{1,3})} \\ &= \frac{(1 + \text{sgn}[(1-2) \cdot (1-3)]) + (1 + \text{sgn}[(1-3) \cdot (3-2)]) + (\dots)}{2 \cdot 3} \\ &= \frac{2}{3} \end{aligned}$$

 风清云的硕士生涯

这里需要注意的地方就是 $\pi(1)=1, \pi(2)=2, \pi(3)=3$ ，而 $\pi_l(1)=1, \pi_l(2)=3$ （即doc 2被放到了位置为3的地方）， $\pi_l(3)=2$ 。因此计算公式如上图所示。这个公式主要想表达的含义在于，如果序列 $\pi$ 中doc1排在doc2之前，在最优序列 $\pi_l$ 中doc1也排在doc2之前的话，sgn函数就会得出1，因此RC值也会相应地变大，而两者顺序相反sgn函数会输出0，RC值减小，因此当序列 $\pi$ 越接近最优序列 $\pi_l$ 的时候，RC的值也会越大。

再计算 $\pi=(1,2,3)$ 和 $\pi_l=(3,1,2)$ 可以得出1/3的结果，取最大可知 $RC=2/3$ 。

## 总结

1，上述评价指标的优点主要是会针对每个query都计算一个值然后取平均，这样某些比较偏激的query的计算结果不太好的时候，也能比较正确的评估模型。

2，上述评价指标的缺点就是针对每个query的计算都是离散的，无法针对参数进行求导。