负样本为王：评Facebook的向量化召回算法

Original 石塔西 推荐道 2020-07-30

有人的地方就会有江湖，就会有鄙视链存在，推荐系统中也不例外。排序、召回，尽管只是革命分工不同，但是我感觉待遇还是相差蛮大的

- 排序
  - 排序，特别是精排，处于整个链条的最后一环，方便直接对业务指标发力。比如优化时长，只要排序模型里，样本加个权就可以了。
  - 排序由于候选集较小，所以有时间使用复杂模型，各种NN，各种Attention，能加的，都给它加上，逼格顿时高大上了许多。
  - 用于排序的基础设施更加完善，特征抽取、指标监控、线上服务都相对成熟。
- 召回
  - 一个系统中不可能只有一路召回，平日里相互竞争，召回结果重合度高，导致单路召回对大盘的影响有限；好处是，相互冗余，有几路出问题，也不至于让排序饿肚子。
  - 召回处于整体推荐链条的前端，其结果经过粗排、精排两次筛选，作用于最终业务指标时，影响力就大大减弱了。
  - 受限于巨大的候选集和实时性要求，召回模型的复杂度受限，不能上太复杂的模型。平日里，最简单的基于统计的策略，比如ItemCF/UserCF表现就很不错，改进的动力也不那么急迫。
  - 用于召回的基础设施不完善。比如，接下来，我们会提到，召回建模时，需要一些压根没有曝光过的样本。而线上很多数据流，设计时只考虑了排序的要求，线上的未曝光样本，线下不可能获取到。

总之，我感觉，排序更受关注，很多问题被研究得更透彻。而召回，尽管"召回不存，排序焉附"，地位相当重要，但是受关注少，有好多基本问题还没有研究清楚。比如接下来，我们要谈到的，诸如“拿什么样的样本训练召回模型”这样的基本问题，很多人还存在误区，习惯性照搬排序的方法，适得其反。

在这种情况下，2020年Facebook最新的论文《Embedding-based Retrieval in Facebook Search》（EBR）更显难能可贵，值得每个做召回算法的同行仔细阅读。正所谓“实践出真知”，文中涉及的部分问题，哪怕你有推荐算法经验但只做过排序（比如一年前的我），都压根不会意识到，更不能给出解决方案。而在你实践过召回算法，特别是向量化召回之后，方能感觉到这篇文章切中召回工作中的痛点，“对症下药”。

所以，今天我将解读一下Facebook EBR这篇经典论文。论文解读不是简单的翻译（这年头，谁还看不懂个英文呢?），而是结合自己的亲身实践，把论文中一笔带过的地方讲透，补充论文中忽略的细节，列举针对相同问题业界其他的解决方案，算是论文原文的一个补充。

简述Facebook EBR模型

这篇论文非常全面，涵盖了一个召回从样本筛选、特征工程、模型设计、离线评估、在线Serving的全流程。但在我看来，并非每部分都是重点。论文中的某些作法就是召回算法的标配。在详细论文EBR的重点之前，我还是将这个模型简单描述一下。

- 模型
  - 老生常谈的双塔模型。双塔避免底层就出现特征交叉，方便拆分模型使doc embedding进入FAISS。
  - 双塔模型不是唯一的，只要<user,doc>的匹配得分能够表达成user embedding/doc embedding内积或cosine的形式，比如FM，都适用于召回。
- LOSS
  - 文中了使用了Pairwise Hinge Loss的形式，loss=max(0,margin-<u,d+>+<u,d->)。即同一个用户与“正文章”（点击过的文章）的匹配度<u,d+>，要比用户与“负文章”（怎么选择负文章就是召回的关键）的匹配度<u,d->高于一定的阈值
  - 尽管不同召回算法有不同的loss，但是背后基于Pairwise LTR的思想都是共通，这一点与排序时采用binary cross entropy loss有较大的不同。
    - 比如Youtube/DSSM模型使用(Sampled) Softmax。但经过Negative Sampling之后，同样是针对同一个用户，一个d+要配置若干个d-，与Pairwise思路类似
    - 而想让u在d+上的概率最高，同样要求分子<u,d+>尽可能大，而分母所有的<u,d->尽可能小，与LTR的思路类似。
  - 但是，根据我的实践经验，使用BPR loss=log(1+exp(<u,d->-<u,d+>))，效果要比这里的Pairwise Hinge Loss好
    - 文中也说了margin对于模型效果影响巨大，BPR loss少了一个需要调整的超参
    - Hinge loss中，<u,d+>-<u,d->大于margin后，对于loss的贡献减少为0。而BPR则没有针对<u,d+>-<u,d->设定上限，鼓励优势越大越好。
- 离线评估
  - 由于线上AB测度，周期长，易受外界影响。因此，在线上AB测试之前，我们需要进行充分的离线评估，减少实验的时间成本。
  - 文中所使用的方法是，拿Top K召回结果与用户实际点击做交集，然后计算precision/recall
  - Airbnb所使用的方法是看“用户实际点击”在“召回结果”中的平均位置

    - 实际上不同于排序，召回的离线评测更加难做，因为召回的结果未被用户点击，未必说明用户不喜欢，而可能是由于该召回结果压根没有给用户曝光过
    - 根据我的亲身实践，无论是precision/recall还是平均位置，和实际线上结果之间还是存在一定gap。曾经发生过，新模型的离线评测结果没有显著提升，但是上线后去发现提升明显。所以，缺乏置信度高的离线评测手段仍然是召回工作中的痛点。
- 特征：因项目而异，可借鉴的地方不多
- FAISS调优：毕竟与Faiss师出同门，非常宝贵的调参经验。但是受篇幅所限，就不在这里过多展开了。

重中之重是"筛选（负）样本"

简单描述之后，才进入本文的华彩段落，即Training Data Mining。单单Mining一个词，感觉就比Sample Filtering高大上了许多，也让你感觉到这件工作的艰巨，远不是“过滤脏数据”那么简单。本文最精彩的部分就是对样本的筛选。如果说排序是特征的艺术，那么召回就是样本的艺术，特别是负样本的艺术。样本选择错了，那么上述的模型设计、特征工程，只能是南辕北辙，做得越卖力，错得越离谱。本文在“样本筛选”上的贡献有三：

- 破除“召回照搬排序”的迷信，明确指出，不能（只）拿“曝光未点击”做负样本
- 提出了easy negative/hard negative的样本分级思路
- 提出了增强hard negative的思路（也不算首创了，和百度Mobius的思路几乎一模一样，也算英雄所见略同了）

本文的另外两个贡献也算是“样本筛选”的引申：

- 全链路优化中提出将召回结果交给人工审核，不过是增强hard negative的一个手段
- 多模型整合，也算是hard negative除了“增强样本”之外的另一种利用方式。而且不同模型也是按照negative samples的难度来分级的

为什么不能（只）拿"曝光未点击"做负样本

我曾经做过Youtube召回，当时就特别不理解为什么Youtube不用“曝光未点击”做负样本，而是拿抽样结果做负样本。而且这样做的还不仅仅Youtube一家，Microsoft的DSSM中的负样本也是随机抽取来的。两篇文章都没说明这样选择负样本的原因。

当时，我只有排序方面的经验，而排序是非常讲究所谓的“真负”样本的，即必须拿“曝光未点击”的样本做负样本。为此，还有所谓above click的作法，即只拿点击文章以上的未点击文章做负

样本。所以，排序思维根深蒂固的我，觉得拿“曝光未点击”做负样本，简直是天经地义，何况还那么有诱惑力

- “曝光未点击”是用户偏好的“真实”（cross finger）反馈。而随机抽取的，可能压根就没有曝光过，不能断定用户就一定不喜欢。
- “曝光未点击”的数量有限，处理起来更快
- 用“曝光未点击”数据，能够复用排序的data pipeline，很多数据都已经处理好了，无须重新开发和生成
- 碰上“万能”的FM模型，拿曝光数据训练出来的模型，既能做排序，又能做召回，岂不美哉😳

所以，当我第一次实践Youtube算法时，直接拿“曝光未点击”样本做负样本，训练出来的模型，离线AUC达到0.7+，做为一个特征简化的召回模型，已经算是非常高了，但是线上表现一塌糊涂。离线评测时，发现召回的物料与用户画像、用户点击历史，完全没有相关性。而当我抱着“照着论文画瓢”拿随机采样的样本做负样本，线上结果却非常好。

何也？？？其实，是因为我自以为是的做法，违反了机器学习的一条基本原则，就是**离线训练数据的分布，应该与线上实际应用的数据，保持一致**。

以前，我们谈到召回与排序的不同，往往只强调速度，即因为召回的候选集更大，所以要求速度更快，所以模型可以简单一些。这是只知其一。另一个方面，起码之前是我没有深刻意识到的，就是

- 排序其目标是“从用户可能喜欢的当中挑选出用户最喜欢的”，是为了优中选优。与召回相比，排序面对的数据环境，简直就是**温室里的花朵**。
- 召回是“是将用户可能喜欢的，和海量对用户根本不靠谱的，分隔开”，所以召回在线上所面对的数据环境，就是**鱼龙混杂、良莠不齐**。

所以，要求喂入召回模型的样本，既要让模型见过最匹配的<user,doc>，也要让模型见过最不靠谱的<user,doc>，才能让模型达到"开眼界、见世面"的目的，从而在“大是大非”上不犯错误。

- 最匹配的<user,doc>，没有异议，就是用户点击的样本
- 最不靠谱的<user,doc>，是“曝光未点击”样本吗？显然不是。这里牵扯到一个推荐系统里常见的bias，就是我们从线上日志获得的训练样本，已经是上一版本的召回、粗排、精排替用户筛选过的，即已经是对用户“比较靠谱”的样本了。拿这样的样本训练出来的模型做召回，一叶障目，只见树木，不见森林。

就好比：一个老学究，一辈子待在象牙塔中，查古籍，访高人，能够一眼看出一件文物是“西周”的还是“东周”的。但是这样的老学究，到了潘家园，却很可能打眼，看不出一件文物是“上

周”的。为什么？他一辈子只见过“老的”和“更老的”，“新的”？被象牙塔那阅人无数的门卫给屏蔽了，他压根就没见过。

拿随机采样做负样本

所以文章中描述的基本版本就是拿点击样本做正样本，拿随机采样做负样本。因为线上召回时，候选库里大多数的物料是与用户八杆子打不着的，随机抽样能够很好地模拟这一分布。

但是文章中没有说明随机抽样的概率，千万不要以为是在整个候选库里等概率抽样。

- 在任何一个推荐系统中，“八二定律”都是不可避免的，也就是少数热门物料占据了绝大多数的曝光与点击
- 这样一来，正样本被少数热门物料所绑架，导致所有人的召回结果都集中于少数热门物料，完全失去了个性化。
- 因此，当热门物料做正样本时，要降采样，减少对正样本集的绑架。比如，某物料成为正样本的概率如下，其中z(wi)是第i个物料的曝光或点击占比
  - $$P_{pos} = (\sqrt{\frac{z(w_i)}{0.001}} + 1)\frac{0.001}{z(w_i)}$$
- 当热门物料做负样本时，要适当过采样，抵销热门物料对正样本集的绑架；同时，也要保证冷门物料在负样本集中有出现的机会。比如，某物料成为负样本的概率如下，其中n(w)是第i个物料的出现次数，而一般取0.75
  - $$P_{neg} = \frac{n(w_i)^{\alpha}}{\sum_j n(w_j)^{\alpha}}$$

NLP背景的同学看以上两个采样公式是不是有点眼熟？没错，它们就是word2vec中所采用的采样公式。没错，word2vec也可以看成一个召回问题，由center word在整个词典中召回context word。

但是，使用随机采样做负样本，也有其缺点，即与d+ 相比，d- 与user太不匹配了。这样训练出来的模型，只能学到粗粒度上的差异，却无法感知到细微差别。就好比，一个推荐宠物的算法，能够正确做到向爱狗人士推荐狗，向爱猫人士推荐猫，但是在推荐狗的时候，无法精确感受到用户偏好上的细微差别，将各个犬种一视同仁地推出去。这样的推荐算法，用户也不会买账。

挖掘Hard Negative增强样本

将的匹配度分成三个档次

- 匹配度最高的，是以用户点击为代表的，那是正样本。
- 匹配度最低的，那是随机抽取的。能被一眼看穿，没难度，所谓的easy negative，达不到锻炼模型的目的。
- 所以要选取一部分匹配度适中的，能够增加模型在训练时的难度，让模型能够关注细节，这就是所谓的hard negative。

如何选取hard negative，业界有不同的做法。Airbnb在《Real-time Personalization using Embeddings for Search Ranking at Airbnb》一文中的做法，就是根据业务逻辑来选取hard negative

- 增加与正样本同城的房间作为负样本，增强了正负样本在地域上的相似性，加大了模型的学习难度
- 增加“被房主拒绝”作为负样本，增强了正负样本在“匹配用户兴趣爱好”上的相似性，加大了模型的学习难度

当业务逻辑没有那么明显的信号时，就只能依靠模型自己来挖掘。这也是本文与百度Mobius的作法，二者的作法极其相似，**都是用上一版本的召回模型筛选出"没那么相似"的对，作为额外负样本，训练下一版本召回模型。**

- 怎么定义“没那么相似”？文章中是拿召回位置在101~500上的物料。排名太靠前那是正样本，不能用；太靠后，与随机无异，也不能用；只能取中段。
- 上一个版本的召回模型作用于哪一个候选集上？文章中提供了online和offline两个版本。online时就是一个batch中所有user与所有d+ 的cross join，这一点就与Mobius几乎一模一样了。
- offline的时候，需要拿上一版本的召回模型过一遍历史数据，候选集太大，需要用到基于FAISS的ANN。

可能有人还有疑问，这样选择出来的hard negative已经被当前模型判断为“没那么相似”了，那拿它们作为负样本训练模型，还能提供额外信息吗？能起到改善模型的作用吗？我觉得，

- 一来，这是一个“量变”变“质变”的过程。在上一版召回模型中，这批样本只是“相似度”比较靠后而已；而在训练新模型时，直接划为负样本，从“人民内部矛盾”升级为"敌我矛盾"，能够迫使模型进一步与这部分hard negative“划清界限”

- 二来，毕竟是百度和Facebook两家团队背书过的方案，还是值得一试。

不过需要特别强调的是，hard negative并非要替代easy negative，而是easy negative的补充。在数量上，负样本还是以easy negative为主，文章中经验是将比例维持在easy:hard=100:1。毕竟线上召回时，库里绝大多数的物料是与用户八杆子打不着的easy negative，保证easy negative的数量优势，才能hold住模型的及格线。

不同难度的模型融合

上一节只是从“样本增强”的角度来利用hard negative。还有一种思路，就是用不同难度的negative训练不同难度的模型，再做多模型的融合。

并行融合

不同难度的模型独立打分，最终取Top K的分数依据是多模型打分的加权和（各模型的权重是超参，需要手工调整）：

$$S_w(q,d) = \sum_{i=1}^{n} \alpha_i cos(V_{q,i}, U_{*,*})$$

但是线上召回时，为了能够使用FAISS，必须将多个模型产出的embedding融合成一个向量。做法也非常简单，将权重乘在user embedding或item embedding一侧，然后将各个模型产出的embedding拼接起来，再喂入FAISS。这样做，能够保证拼接向量之间的cosine，与各单独向量cosine之后的加权和，成正比。

$$E_Q = (\alpha_1 \frac{V_{Q,1}}{\|V_{Q,1}\|}, \cdots, \alpha_n \frac{V_{Q,n}}{\|V_{Q,n}\|})$$

$$E_D = (\frac{U_{D,1}}{\|U_{D,1}\|}, \cdots, \frac{U_{Q,n}}{\|U_{Q,n}\|})$$

$$\cos(E_Q, E_D) \overset{\text{成比例}}{=} \frac{S_w(Q,D)}{\sqrt{\sum_i^n \alpha_i^2} \cdot \sqrt{n}}$$

固定的

easy model肯定是拿随机采样训练出来的模型，这个没有异议。问题是hard model是哪一个?

- 文章中指出，使用“曝光未点击”作为hard negative训练出来的hard model，离线指标好，但是线上没有效果
- 反而，使用挖掘出来的hard negative训练出来的hard model，也easy model融合的效果最好。

串行融合

其实就是粗排，候选物料先经过easy model的初筛，再经过hard model的二次筛选，剩余结果再交给下游，更复杂粗排或是精排。

根据文章中的经验，使用“曝光未点击”作hard negative训练出来的hard model同样没有效果，反而是挖掘出来的hard negative训练出来的hard model做二次筛选更加有效。

全链路优化

这一节切切实实cover到了我们召回工作中的痛点，就是新的召回算法，往往不受ranker的待见

- ranker是在已有召回算法筛选过的数据的基础上训练出来的，日积月累，也强化了某种刻板印象，即<user,doc>之间的匹配模式，只有符合某个老召回描述的匹配模式，才能得到ranker的青睐，才有机会曝光给用户，才能排在容易被用户点击的好位置。
- 新召回所引入的<user,doc>之间的匹配模式，突破了ranker的传统认知，曝光给用户的机会就不多。即使曝光了，也会被ranker排在不讨喜的位置，不太容易被用户点击。
- 这就形成一个恶性循环，新召回被ranker歧视，后验指标不好看，让ranker歧视的理由更加充分，然后曝光的机会更小，排到的位置更差。

但是本文针对这一问题的解决方案有限：

- 将各路召回的打分作为特征，加入到训练ranker的过程中。但是，这改变不了新召回是少数派的现实，新召回的打分作为特征在训练样本中的占比有限，发挥的作为也可能有限
- 将召回结果交人工审核，发现bad case，增强下一版本的训练样本。这就是一种人工寻找hard negative的方式，不太现实。

总而言之，各路召回之间的重叠、竞争，召回与排序之间的“相爱相杀”，将会是推荐系统内一个无解的永恒话题。

"曝光未点击"就是鸡肋

最后说一下“曝光未点击”样本。在排序中，“曝光未点击”是要被严格筛选的“真负”样本，对于排序模型的成败，至关重要。而到了召回中，无论是根据我自己的亲身实践还是Facebook的经验，“曝光未点击”样本都是妥妥的鸡肋，“食之无味，弃之可惜”

- 前面说了，“曝光未点击”不能单独拿来做负样本。
- “曝光未点击”作为hard negative增强训练集，和easy negative一起训练出一个模型，亲验线上没任何提升。
- “曝光未点击”毕竟经过召回、粗排、精排，说明与用户还是比较匹配的，能否作为正样本？我也有过类似的想法，但是Facebook的论文告诉我，没啥效果。（“曝光未点击”能否作为权重较小的正样本？）
- Facebook论文指出，拿“曝光未点击”训练出来的hard model，与easy model无论是串行融合还是并行融合（粗排），都没啥效果，这也被我们的实验结论所印证。

导致“曝光未点击”成为鸡肋的原因，可能是由于它的两面性：

- 它们经过上一轮召回、粗排、精排的层层筛选，推荐系统以为用户会喜欢它们。
- 而用户的行为表明了对它们的嫌弃。（当然用户“不点击"也包含很多噪声，未必代表用户不喜欢）

“曝光未点击”样本的两面性，使它们既不能成为合格的正样本，也不能成为合格的负样本。唉，“大是大非”面前，“骑墙派”总是不受欢迎的😕

总结

很高兴能看到一篇论文，契合了你之前的实践和经验，不由得生出“英雄所见略同”的慨叹，特别是在召回这一重要但是受关注较少的领域。

我给本文起名为"**负样本为王**"，就是想传递这样的观点：如果说排序是特征的艺术，那么召回就是样本的艺术，特别是负样本的艺术。负样本的选择对于召回算法的成败是决定性的，

- 选对了，你就hold住了及格线。之后的模型设计、Loss设计、特征工程、Serving优化不过是锦上添花。
- 选错了，之后的模型设计、Loss设计、特征工程、Serving优化都是南辕北辙，无本之木，平添无用功罢了。

至于怎么选对样本，Facebook的这篇论文是难得的来自一线实践的经验之谈，值得每个做召回的算法同行认真阅读。