向量体系(Embedding)在严选的落地实践

原创 严选技术 严选技术团队 3天前

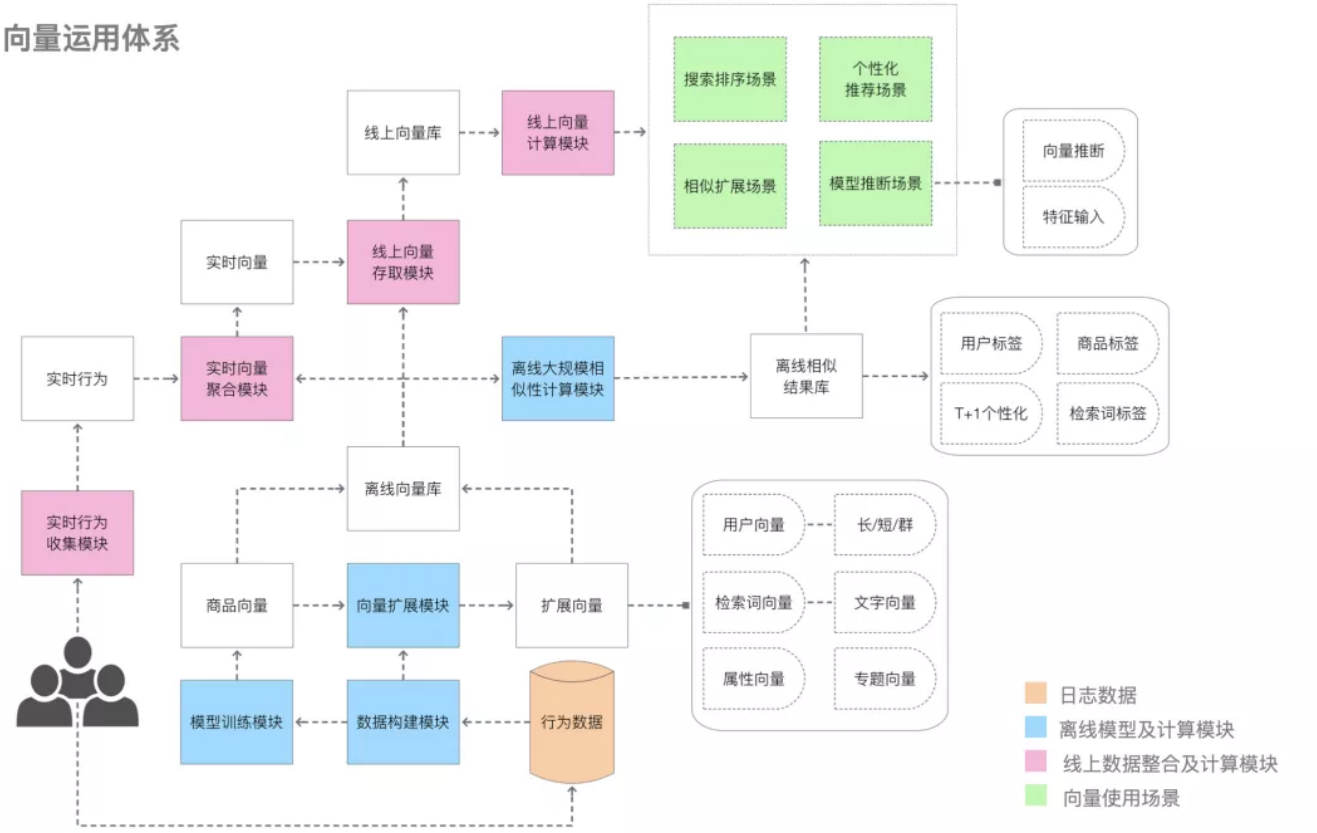
向量化在业界的运用越来越广，近期也有许多文章分享过相关的主题。严选于18年下半年开始探索向量化在搜索推荐场景中的运用，从最开始基于商品召回用户的任务到后续的搜索召回、搜索个性化排序、搜索底纹、搜索发现词、搜索建议词、跨类目推荐、推荐召回、多兴趣召回、通用排序、端智能重排等等，我们不断拓宽向量体系在严选的运用，在这过程中一点点迭代与沉淀。本文将从模型算法和落地运用等角度做简要介绍，希望能给读者一些启发。

本文将从以下几个方面展开介绍

向量体系 | 算法模型 | 相关技术 | 落地分享 | 总结展望

篇幅较长，时间充裕的读者建议全文阅读，不充裕的话也可以有选择阅读。

向量体系



上图是对严选向量体系的一个概览。引言中说了那么多运用场景，第一眼看会觉得有些场景之间跨度蛮大，但是仔细考虑一下我们会发现，其实电商场景的大部分任务（包含以上所述的）都是在做对象之间的匹配，可能是商品和人的匹配，可能是检索词和商品的匹配，可能是用户和检索词的匹配等等。

于是，摆在我们面前的问题就是如何把我们目前遇到的以及将来会遇到的对象进行一个比较好的表征并刻画对象之间的相似度，其本质是学习出各个对象在同一个空间中距离的刻画。

如果我们把各个对象都学习到同一个空间，那么这些对象之间的组合几乎能覆盖各种运用场景，例如我们拥有用户(U)、商品(I)、检索词(Q) 的表征，仅仅是这三者的组合就可以有例如U2I、Q2I、I2Q、Q2Q、U2Q、I2I等等，我们进一步扩展类目、专题等等的表征之后能覆盖的场景就更多了。

那么我们如何去表征各个对象呢？向量凭借着简单的结构、快速的相似性计算、强大的表征能力有着得天独厚的优势。因此我们选择向量作为对象的表征方式。

当然有了向量表征仅仅是第一步，为了能让其服务于各个场景，线上线下需要其他模块的辅助配合，需要将具体的场景进行抽象，同时过程中也离不开不断的迭代优化，需要考虑模型的效果、稳定性、可扩展性、以及线上性能等等。

由点及面，在有了向量基础后，我们需要进一步扩展向量存储、向量计算等能力，进一步，由面到体，我们需要不断去分析和抽象具体的业务场景，以此不断铺开向量体系在实际业务中的落地运用。

算法模型

最开始我们使用的是图嵌入技术同时学习商品和用户，例如LINE、Node2Vec等模型（节点做标示进行区分），后续也尝试过使用YoutubeDNN的方式学习商品和用户向量，最后为了兼顾模型的可扩展性以及稀疏数据上表征的可靠性，我们选择了两步走的策略。

1. 确认商品是整个电商场景中的核心，单独学习优化商品向量表征

2. 其他对象都和商品有直接或者间接的关系（交互），通过专门的聚合模块得到对象表征

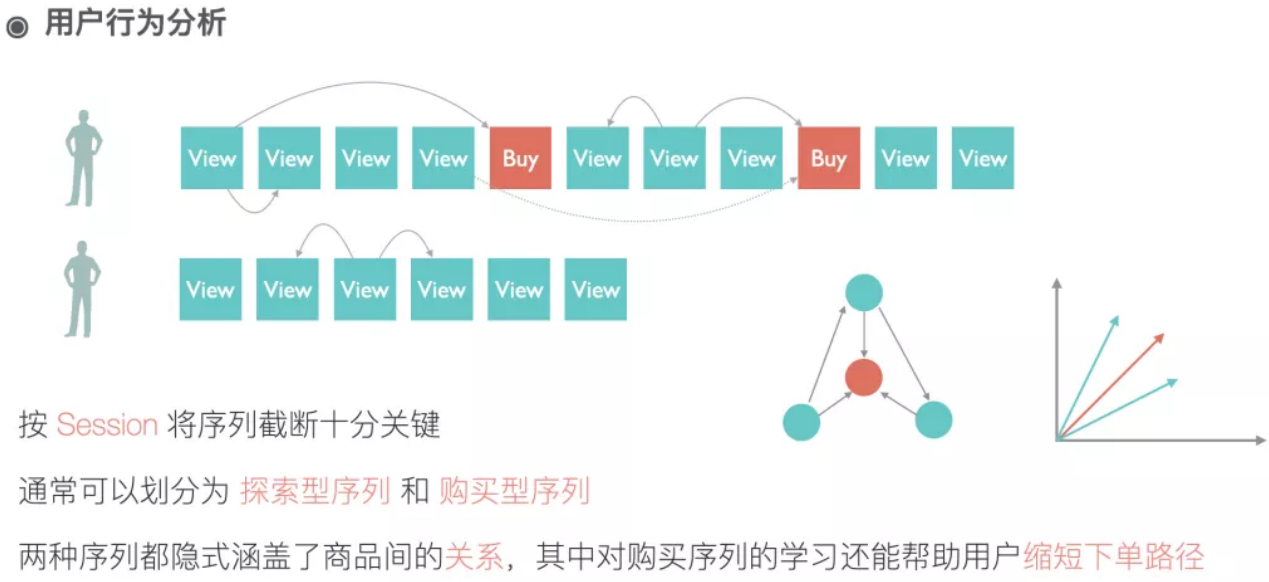
实践表明，这个两步走的策略在我们的场景中要优于之前尝试的方案，接下来我会对这两步做简要的介绍。

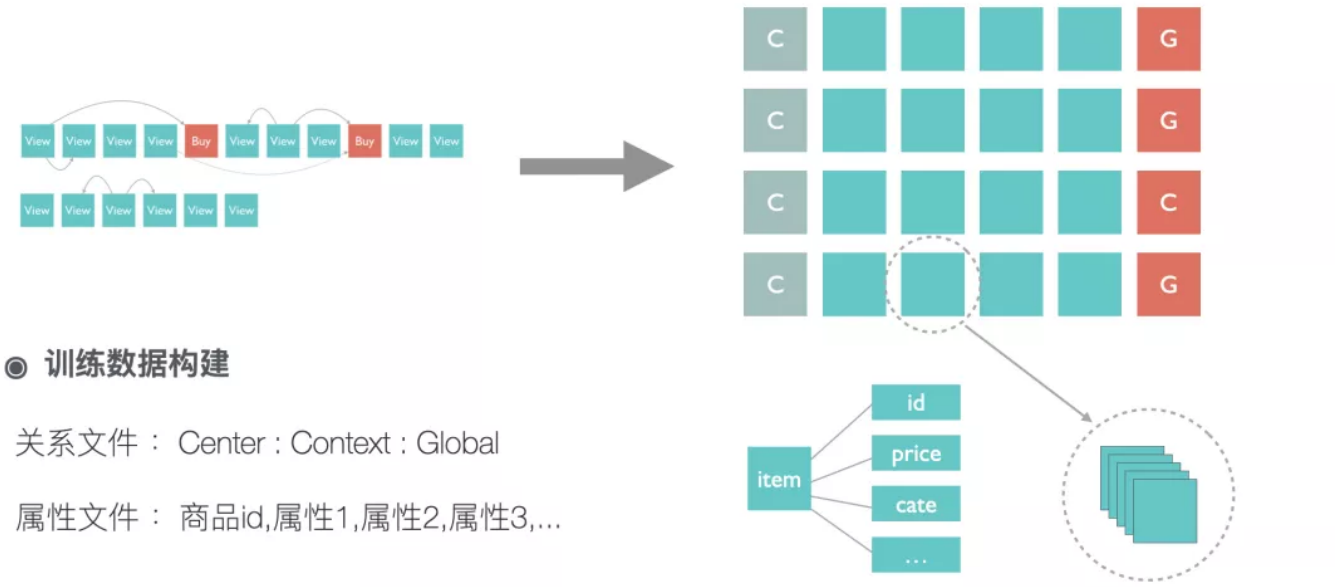
商品向量学习

商品向量的学习关键在于用户行为数据的梳理以及模型的构建（参考了Airbnb的模型，在其基础上做了较多改进）。

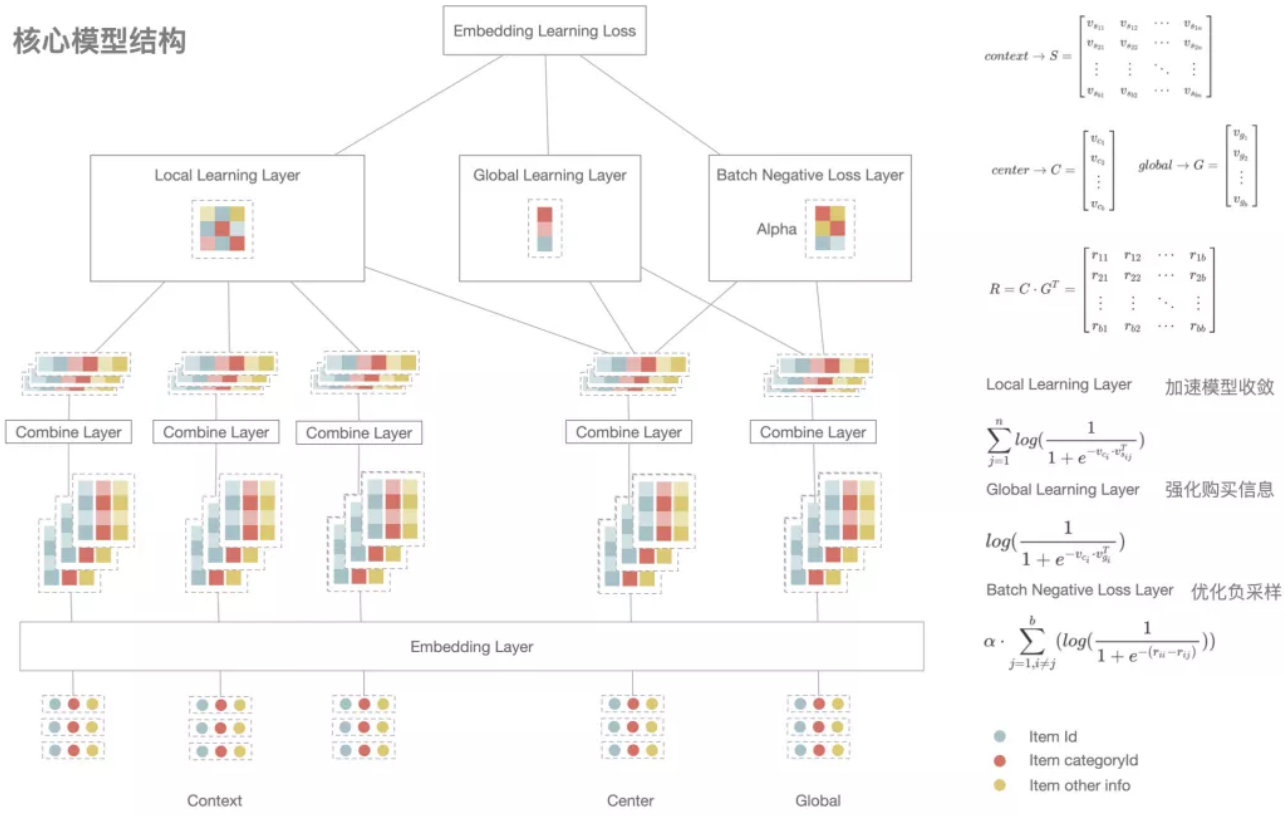
商品是电商场景下用户交互最核心的东西，大部分的用户行为都围绕着商品。大量的用户行为中，天然的包含了用户对商品的认知，通过对这些认知的提取，我们便能够刻画出用户眼中商品的样子，利用用户眼中商品的样子能更好得去引导用户的行为。

我们有用户和商品交互的各种数据以及商品自身的一些属性。用户的**连续点击行为**能在商品之间构建关联；用户的**购买行为**能告诉我们用户的探索路径更容易收敛到哪些商品；用户的相继的购买行为能反应商品之间的搭配购买信息；商品自身的**属性**能在冷启动时给我们提供很多额外信息等等。我们要做的就是融合**行为**和**属性**学习出商品向量。





从上图可以看到，我们训练数据的构建其实可以等效看成构建了商品之间有权有向的关系图，同时增加了一些转化导向的长连接。我们的模型训练有别于传统的方式，对一个单一训练样本 center(中心商品):contexts(临近点击):global(序列内购买)，我们希望 center 和 contexts(包含多个id) 以及 center 和 global 的距离越接近越好，所以在构建损失函数的时候我们一次性算出 center 和 contexts(包含多个id) 的相似性以及 center 和 global 的相似性，然后再加上负采样的约束。对于负采样，我们采取的是在一个 batch 内构建负样本的方式，这种方式简单有效且能满足按照样本出现的频率采样，同时结合矩阵运算能加快训练数据。在获取负样本后我们进一步构建 pair-wise 的 loss，这对模型效果能有一些提升。模型的细节可以参考下图。



$$R = C \cdot G^T = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1b} \\ r_{21} & r_{22} & \cdots & r_{2b} \\ \vdots & \vdots & \ddots & \vdots \\ r_{b1} & r_{b2} & \cdots & r_{bb} \end{bmatrix} \qquad R_{diag} = \begin{bmatrix} r_{11} \\ r_{22} \\ \vdots \\ r_{bb} \end{bmatrix} \qquad R_{diff} = R_{diag} - R = \begin{bmatrix} 0 & r_{11} - r_{12} & \cdots & r_{11} - r_{1b} \\ r_{22} - r_{21} & 0 & \cdots & r_{22} - r_{2b} \\ \vdots & \vdots & \ddots & \vdots \\ r_{bb} - r_{b1} & r_{bb} - r_{b2} & \cdots & 0 \end{bmatrix}$$

◉ **模型 loss**

$$-\sum_{i=1}^{i=b}(\sum_{j=1}^{n} log(\frac{1}{1+e^{-v_{c_i} \cdot v_{s_{ij}}^T}}) + log(\frac{1}{1+e^{-v_{c_i} \cdot v_{g_i}^T}}) + \alpha \cdot \sum_{j=1, i \neq j}^{b} (log(\frac{1}{1+e^{-(r_{ii}-r_{ij})}})))$$

$$-\sum_{i}(\sum_{j}(log(\sigma(Squeeze(C_{expand} \cdot S^T))) + log(\sigma(diag(C \cdot G^T))) + \alpha \cdot \sum_{j}(log(\sigma(R_{diff})))))$$

还有需要提及的一点是，为了学习到更好的商品向量表征，通常的方式是每日全量重新训练商品向量，但是这就引入了一些问题，例如每日的商品向量不在同一个空间中，相互之间的计算是没有意义的，如果不小心拿隔日向量进行了计算，会引入意外的结果。于是有人会采用增量训练的方式，每日只对新增的商品做推断获取它们的向量，但是这样也会引入一些问题，比如商品之间的关系并不是稳定不变的，只针对新商品做推断的话会使得原有商品之间的关系无法依据新数据进行调整。

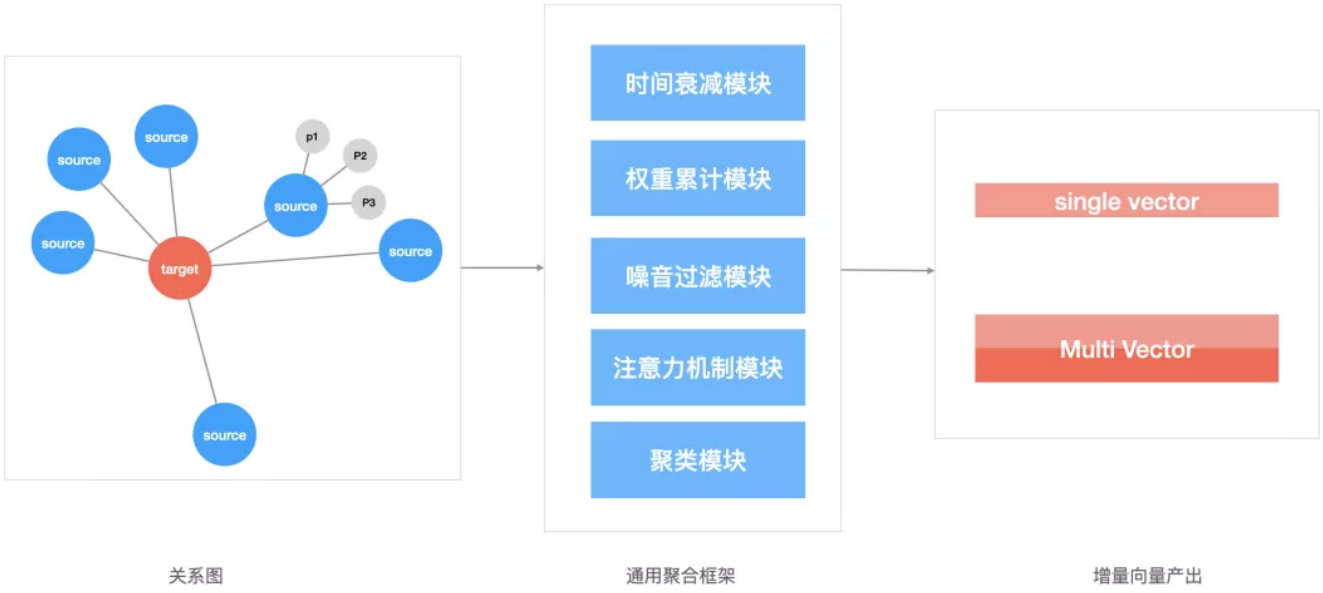
针对这些问题我们提出了两种解决方案，一种是在隔日的向量空间之间学习一个仿射变换，对当日学得的向量做仿射变换，然后最小化相邻两天共现对象向量间的差异，这样仿射变换后的向量就能变换到历史空间中，同时包含一些新的结构信息。还有一种方案类似预训练的方式，在第二日训练的时候拿前一日的向量做初始化，对于没有出现过的对象做随机初始化，然后降低学习率做微调学习。在我们的场景中第二种方案效果相对好一些，第一种也能取得不错的效果。

此外从19年下半年开始我们也针对GNN相关的模型做了尝试，例如graphSAGE、LightGCN、SR-GNN等，目前看来SR-GNN模型在我们的场景中效果还可以。

扩展向量学习

假设我们只有商品向量(I)，我们能做的事相对比较局限，例如我们可以做基于I2I的各种任务、我们可以将商品向量作为预训练的向量提供给具体场景用于迁移学习。进一步我们获取了用户向量(U)之后，我们便可以做U2I、I2U等召回场景，很多分享案例都止步于此，但是只要我们再加一个对象例如检索词向量(U)，我们的运用场景就瞬间开阔了许多，进一步我们还会有类目向量(C)、专题向量(T)等等。

随着业务场景的铺开，我们会遇到越来越多的对象匹配任务。我们可以针对每个场景分别学习，例如针对商品和检索词构建深度模型，学习他们的相似度量，例如针对检索词历史进行挖掘做检索词的相关推荐，例如针对用户和商品的历史交互建模做商品的召回，例如针对用户和专题的交互历史做专题召回等等，在这样的模式下每新增一个场景我们就需要花费人力去开发和后期维护，并且有的新增场景一开始并没有训练数据给你使用。考虑到可扩展性，我们可以将这些对象统一到一个增量向量产出框架中，抽象出一套聚合逻辑，产出同一向量空间中的向量。



结合落地实践，我们的聚合模块主要包含上图所示的子模块， 我们将新增对象叫做target，将基础的商品叫做source。到实际场景中，对象和商品的交互可能存在时间上的先后，因此需要时间衰减模块来处理时间因子；考虑到对象和商品的交互次数会有频次上的差异，我们需要权重累计模块来处理这个因素；考虑到对象和商品的交互会有一些噪音数据比如用户的误点击行为、或者运营的错误配置等等，我们需要噪音过滤模块来对噪音数据降权；考虑到有的场景使用单向量表征更合适，我们需要对交互数据做自注意力机制的处理，来凸显主要兴趣，因此引入了注意力机制模块；但是在有些场景，对象交互的商品往往是分布在空间中的多个区域（对用户而言是多兴趣表征，对检索词而言是多义词表征），为此我们加入了聚类模块（可以使用传统的聚类算法，也可以考虑用复杂网络中的社区检测算法来进行聚类，一般是在全局商品上进行操作），来输出对象的多向量表征。

凭借通用聚合框架，每次我们要新增对象的向量表征的时候，只需要处理一份target和source的关系表，同时每一个source都带上target和它交互的相关附加信息，将这份关系数据输入聚合框架便能产出和商品向量属于同一向量空间的对象向量了，任意对象之间都可以相互计算，判断相似情况。此外聚合模块的逻辑经过较小的改动也可以直接运用到线上对象的实时向量表征中。

相关技术

要建造大楼仅仅有基础的砖块肯定是不够的，我们还需要钢筋和水泥，需要脚手架等等。

有了基础向量，随之而来的就是大规模向量计算问题。初期各种任务主要集中在离线计算，所以我们自研了基于数据分块、矩阵运算和并行计算的离线大规模相似度计算模块，百亿规模的精确计算在单机上基本能在几分钟内完成，后期进一步调研了一些最近邻搜索算法后，使用了LSH、FAISS等方式来做大规模向量召回，并运用到线上实时召回中，感兴趣的读者可自行查阅相关资料。

我们的向量体系不仅仅运用在常见的召回任务中，在很多线上的基础排序任务中也发挥了重要作用。我们开发了一套线上向量存取和实时向量异步聚合的服务。基于这个服务，我们进一步开发了通用排序服务，例如基于用户的实时向量对搜索结果做Top个性化重排、基于检索词对专题进行排序、对众多的活动页商品做实时个性化排序等等。也正是基于这个服务，我们在一定程度上将搜索和推荐的部分任务统一到了同一个框架中。

此外在电商场景中效果的提升离不开实时数据的辅助，因此在我们的体系中实时行为数据模块也发挥了重要的作用。

落地分享

最后针对一些具体的场景展开介绍下，给读者一个更为完整的阅读体验。

搜索场景

先谈一谈搜索场景，搜索场景不仅仅限于商品的召回和排序，搜索的底纹、发现词、建议词等都能为搜索导流，而这些也都能利用统一向量体系得到较好的解决。

例如发现词场景，我们需要基于用户的行为对用户推荐相关可能想搜的词汇，传统的方式通常基于query挖掘，基于用户先前搜索的词汇给用户推荐更多相关的词汇，但其实从用户浏览过的商品也能推测出用户想搜的商品，而这两个角度很自然得可以通过向量体系中的Q2Q（基于用户的搜索词）和I2Q（基于用户的浏览商品）来得到解决，进一步也可以直接通过U2Q（基于用户的各种行为）来获取 。 **（Q2Q 、I2Q 、U2Q的运用）**

在此多说一句，为了更好的效果，通常需要结合离线数据和实时的数据，例如用户（U）就会有离线长期向量、离线短期向量、实时聚合向量（时间维度）、实时多兴趣向量（空间维度）等区分，同时为了性能提升部分相似性计算也可以挪到离线完成，此外在类似推荐的场景使用非多兴趣向量的时候多样性往往扩散得不够，这时候就会需要I2I（可拆分出相似和相关结果）的帮忙。 **（时空表征的运用）**

建议词场景同理，基于Q2Q便可以，当然实际运用时Q2Q中的第二个Q和第一个Q是不同的，第二个Q需要是质量较好的Q，候选是需要考虑Q的句法结构，例如形容词+名词、名词+名词等，需要考虑Q被大众搜索过的次数、需要考虑Q在搜索后能召回的数量、需要考虑Q在搜索后用户点击的数量等等，这些都可以离线处理好，在检索词向量库的基础上过滤出一份优质检索词向量库。 **（Q2Q的运用）**

此外基于检索词向量做商品的召回扩充其实在一定程度上融合并强化了传统的基于同义词扩充召回、基于SEO扩充召回等等方案，因为检索词向量之间的相似性天然的刻画了基于用户行为的词汇同义性，同时单商品上的SEO词会通过商品关系网扩充到其他商品上，那么基于检索词向量自然能召回那些即使没有相应SEO但是应该有那些SEO的商品。 **（Q2I的运用）**

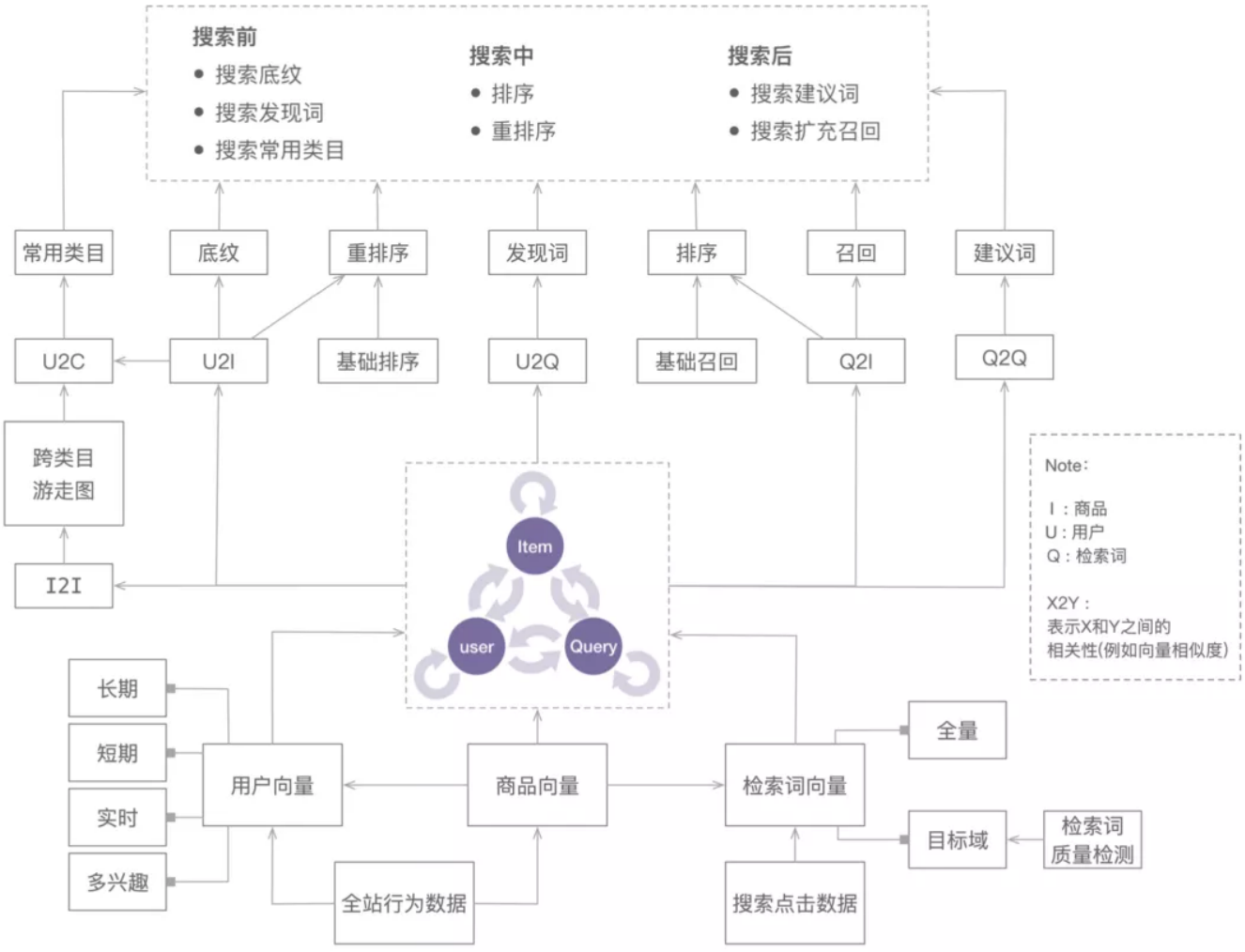
基于向量体系对搜索排序的优化也是水到渠成的，所有的优化可以看作是基于预训练向量的排序运用，而且扩展了数据的边界，不仅仅局限于搜索场景，将全局的行为都融入到了搜索中。

搜索场景的重点是帮用户快速找到他需要的商品，传统的认知是检索词包含了用户所有的意图，但是事实显然不是如此，比如T恤一词就相当宽泛。这其实可以看作是贝叶斯推断，假设我们不知道

用户任何其他信息，基于T恤一词我们会有一个先验估计，得出一个基础排序。**（检索词向量、商品向量的运用）**

但是当我们获取更多是事实之后，我们的信息多了，那么必然可以去修正我们之前估计。例如我知道这个用户前几天看了一些女性鞋子，那么显然将女性T恤排在更前面会是合理的选择**（离线长短期用户向量、商品向量的运用）**，同时我们又知道这个用户刚刚在其他页面浏览了一些运动健身的器械，那么显然将部分运动T恤提前会有不错的收益，实际模型上线后提升了搜索排序的鲁棒性、提高了搜索个性化的实时反馈能力、提升了搜素转化率，取得了不错的效果。**（实时用户向量、商品向量的运用）**

下图是对以上提到的部分场景的一个汇总。



当我们进一步引入类目向量和专题向量后，我们会发现运用的场景又进一步扩充了。在搜索场景中，通常会面临类目预测的问题，当我们拥有了检索词向量和类目向量之后我们能轻松地推断出每个检索词的主类目，对于没有出现过的长检索词，我们在计算前加入分词模块，通常便可推断出他们的类目了。此外当我们设定相似度的阈值之后，我们也可以为检索词作出多类目的预测，以满足不同场景的需求。**（Q2C的运用）**

此外在搜索场景，为了充分利用搜索流量，我们还会做专题的召回以及和商品的混排，商品和专题属于异构数据，通常我们需要构建额外的模型来对专题作出排序。但是当我们有了专题向量之后（此处提一下，专题向量其实可以有两个，一个是基于关联商品聚合的，还有一个是基于专题的文本进行推断得到，因为检索词本质上是文字，专题的文本可以拆分后用检索词向量进行描述），我们便可以方便的对专题进行排序然后呈现。**（Q2T的运用）**

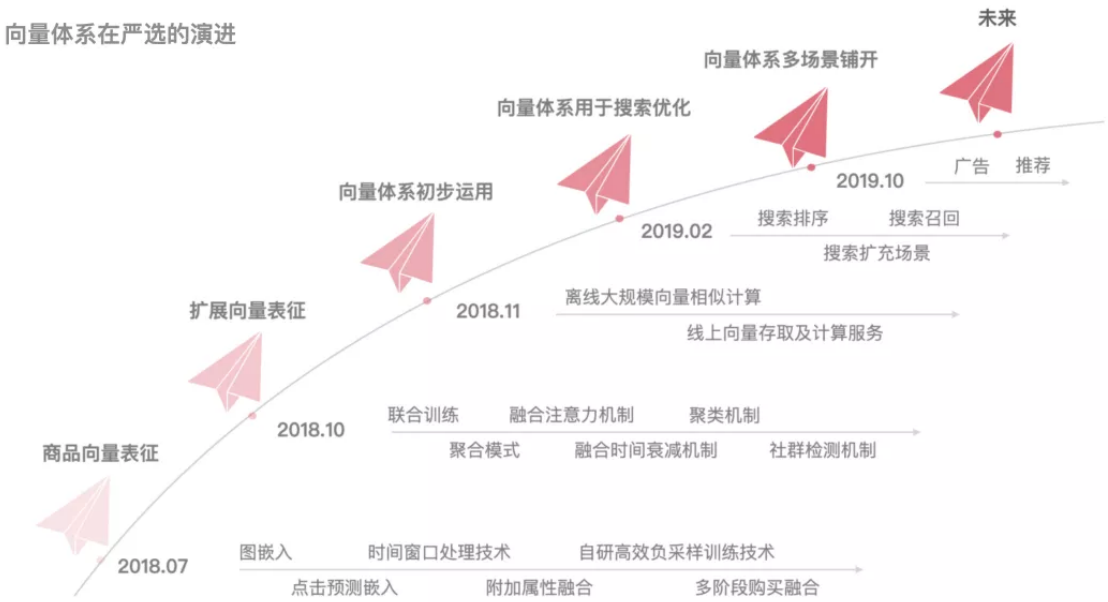
推荐召回

接着我们再说一说向量体系在推荐召回中发挥的作用。这里面涉及到了用户的多种向量表征，对于用户的刻画当然是越精细越好，在我们的场景中我们学习了用户长期向量、短期向量、实时向量、用户多兴趣向量、用户群体向量等等，不同的向量有着不同的使用场景，对于展示位置比较少的推荐位，我们关注用户的主要兴趣，所以一般使用单兴趣向量即可（我们模型训练得到的单兴趣向量也可召回不同兴趣面的商品，但是相似头部的商品相对还是比较集中）；在一些展示位较多的推荐位置例如猜你喜欢模块，我们需要兼顾推荐的多样性，所以一般会使用用户多兴趣向量召回商品（实时多兴趣和离线多兴趣）；对于行为较少的新人，我们则会采用用户群体向量去召回商品，实际中也取得了不错的效果。 **（各种U2I的运用）**

购买预测

此外再提一个购买预测的场景，用户日常的行为可能是目的明确地浏览、也可能是目的不明地闲逛，那么如何区分是哪种行为呢。其实很简单，分析下用户看过的商品之间是相似的还是没什么关联的就行，很显然（I2I）能满足我们的需求，我们可以为每个商品计算它与其它所有商品相似度得分的均值（记为S），如果每个商品最终得分都较大，那么用户是在目的明确地浏览，如果每个商品的得分都较小那么用户是在目的不明地闲逛，如果几个商品得分较高，几个商品得分较低，那么用户是相对集中的看了几个类似商品，同时无意地点了几个其他商品。基于以上我们可以挑选出那些目的明确地浏览的用户。至此读者可能会想，然后把得分最大的商品挑出来就可以了，但是其实还没有结束。截止到目前我们只使用了用户当日的行为，但是我们上手还有用户以往的行为，这些信息不能浪费，它们会修正我们当下作出的判断。假设今日挑选出的商品中，有的商品用户前几日就很有兴趣，那么今日他购买的概率必然比S得分相似的其他商品要更大，因此更好的判断应当基于今日的S以及用户前几日兴趣对商品们的得分（U2I），这个项目的上线明显提升了用户的转化。 **（I2I和U2I的运用）**

总结展望



向量体系有着很明显的优势，它能让我们迅速完成产品新功能的上线并取得不错的效果，同时在有的场景中也能完胜一些老的复杂的方法，这恰恰也印证了奥卡姆剃刀原则。

随着表征对象的增加以及对业务的进一步理解，我们仍将一步步不断拓宽向量体系在业务中的落地，但是在实践中我们也会发现它的局限性，这个局限性一方面来自于向量表征结构自身，另一方

面源于部分实际问题的复杂性需要新的表征方式来解决，所以我们将会进一步探索其他的表征方式，以及新的技术方向。

作者简介

张俊，高级算法工程师，2018年毕业于中国人民大学后加入网易严选 ，致力于严选搜索推荐业务的迭代优化，推动向量体系在严选从无到有的构建并运用于各个业务场景。

END