

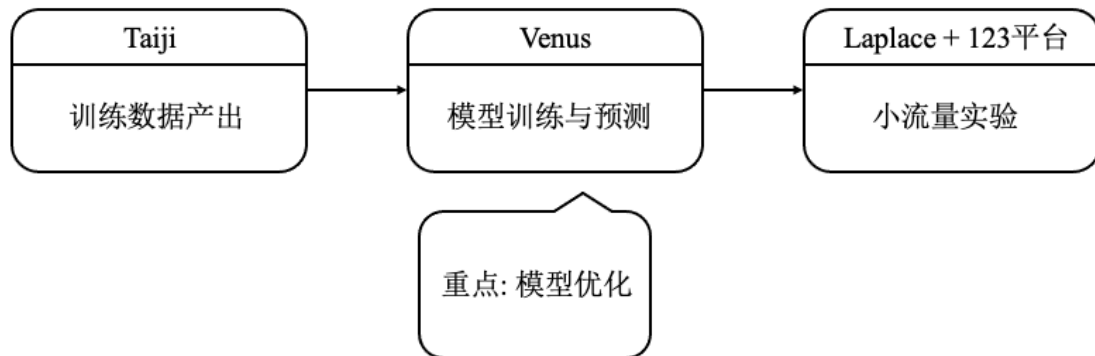
# 实习总结文档

Bingchaowu

## 一、 背景

随着互联网和终端设备的普及，涌现出海量的物品，例如用户拍摄的视频等，推荐系统是从海量的物品中给用户推荐感兴趣的物品。基于内容和基于协同过滤等经典的算法是以一种静态的方式建模用户与物品之间的交互，且捕获用户广义的兴趣。序列推荐系统(SRS)不同于传统推荐系统，它试图理解和建模用户的连续行为、用户与物品之间的交互作用以及用户偏好和商品流行度随时间的变化，通过将用户和商品的交互建模为一个动态的序列并且利用序列的依赖性来活捉当前和最近用户的喜好。在 QQ 小世界的召回场景中，部署序列推荐模型到实际业务场景中颇具挑战。此文档首先介绍序列推荐模型在 QQ 小世界的召回场景中的整体框架，包括产出训练数据、序列模型的训练和预测以及模型上线的流程等，然后依次介绍：1) 产出训练数据包含只有正向行为构成的行为序列和全序列的行为序列；2) Venus 平台上模型训练和预测的流程；3) 上线小流量的具体流程。接着介绍在经典的序列推荐模型基础上的一系列优化策略。最后，介绍离线指标计算的流程用于快速迭代模型和后续一些未尝试的优化点。

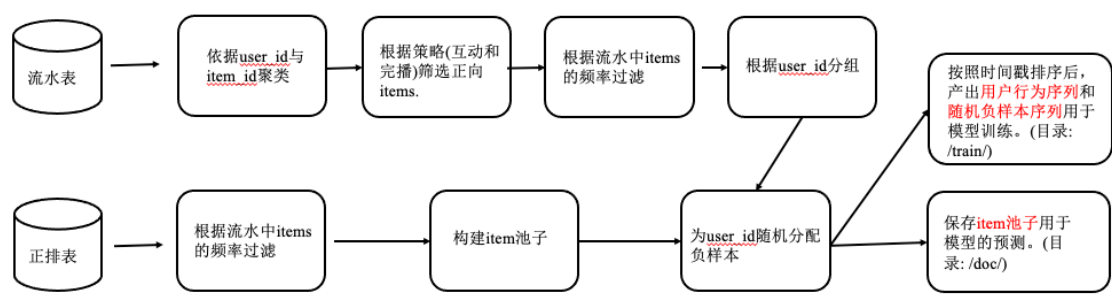
## 二、 序列推荐模型部署框架



序列推荐部署的框架如上图所示，首先太极生成训练数据，包含模型训练的数据和预测数据(item id)。然后在 Venus 平台训练模型和构建模型的预测流程。最后上线小流量实验，只介绍基本的小流量实验上线操作，由于本人对线上流程(CPP 文件)不熟悉，因此不涉及线上的流程，其中重点介绍在 Venus 平台上模型优化的策略。

## 三、 Taiji 产出数据的流程

### 3.1 正向行为序列的训练数据



太极(Taiji)数据产出的整体流程如上图所示，其中输入的是流水表(little\_world\_feed\_flow\_hourly)和正排表(littleworld\_index\_pool)，输出的内容存放在 HDFS 文件系统，具体内容如下：

- 1) train\_no\_hash: 明文的用户行为序列;
- 2) train: 用于训练的 hash 后的用户行为序列，数据格式如下所示:

```
3022739347|1.0|1577396109:1001:1.0;1814462061:1002:1.0;222126691:1003:1.0|1814462061:2001:1.0;222126691:2002:1.0;164826
0295:2003:1.0;613020302:3001:1.0;418490522:3002:1.0;1251270476:3003:1.0;218011836:4001:1.0;982848212:4002:1.0;144990065
0:4003:1.0;4865084773:1101:1.0;5821796598:1102:1.0;4881493681:1103:1.0;5821796598:2101:1.0;4881493681:2102:1.0;51171954
40:2103:1.0;6365432447:3101:1.0;6339498575:3102:1.0;4584470929:3103:1.0;5636498550:4101:1.0;6167520619:4102:1.0;5982905
509:4103:1.0;10221391863:1201:1.0;9553661403:1202:1.0;8914520941:1203:1.0;9553661403:2201:1.0;8914520941:2202:1.0;98237
26248:2203:1.0;8828130581:3201:1.0;10285366628:3202:1.0;9562646478:3203:1.0;8985009524:4201:1.0;10045714256:4202:1.0;99
78911564:4203:1.0
```

其中第一个红框是 user\_id, 10\*\*是 hash 的输入行为序列, 20\*\*是 hash 后的目标行为序列, 是 10\*\*相关序列向右偏移一个 item, 30\*\*是 hash 后的随机负样本序列, 40\*\*是 hash 后的 hard 负样本序列。\*1\*\*和\*2\*\*分别是 cate2 和视频作者相关的序列;

- 3) test\_user\_seq: 从 train 中随机采样部分数据,用于离线评估(计算 HR 和 NDCG);
- 4) test\_target\_item: 将行为序列中 last item 保存作为离线评估中的 ground truth;
- 5) doc: item 池子, 收集指定时间段内的行为序列中出现过的 items;

```
962068710451838990|1.0|658683354:1001:1.0;5241944748:1101:1.0;9489343830:1201:1.0
962262403259564048|1.0|909988113:1001:1.0;5821796598:1101:1.0;8789859995:1201:1.0
962304417418182672|1.0|470376773:1001:1.0;4965829598:1101:1.0;10445484936:1201:1.0
962383230168727567|1.0|1212005171:1001:1.0;6179751162:1101:1.0;10336335941:1201:1.0
```

第一列是明文的 item id, 10\*\*, 11\*\*和 12\*\*分别是 hash 后的 item\_id, cate2\_id 和 author\_id;

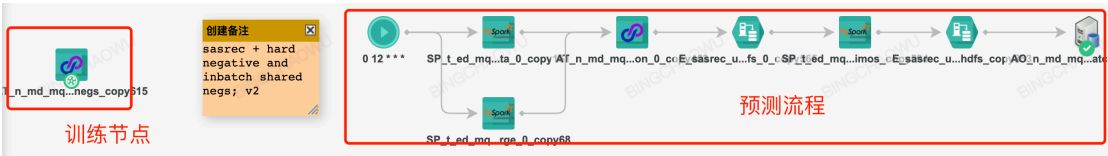
太极输出内容的 HDFS 根目录	hdfs://ss-sng-dc-v2/data/SPARK/SNG/g_sng_im_web/bingchaowu/sasrec/sasrec_sideinfo_v2
太极任务	https://taiji.oa.com/react/index.html#/platform/92404/8108827
源代码	https://git.woa.com/jenesisliu/recall_seq/blob/master/sasrec_u2i_bingchaowu/generate_pos_train_data.py

### 3.2 全序列的训练数据

3.1 节是产出用户交互过的正向 items 构成的序列，其中正向 items 的定义是互动和完播，忽略了其他的 items，例如未满足完播的 items，通过建模用户交互过的全部 items 能更加精确建模用户的动态兴趣。全序列与正向行为序列的差异是构建的基本单元不同，全序列是由“item-level”构成，其中 level 表示不同完播等级，具体定义见源代码，而正向行为序列的基本单元是“item”。

输出内容的 HDFS 根目录	hdfs://ss-sng-dc-v2/data/SPARK/SNG/g_sng_im_web/bingchaowu/sasrec_all_seq
源代码	<a href="https://git.woa.com/jenesislui/recall_seq/blob/master/sasrec_u2i_bingchaowu/generate_all_seq_train_data.py">https://git.woa.com/jenesislui/recall_seq/blob/master/sasrec_u2i_bingchaowu/generate_all_seq_train_data.py</a>
Venus 中作业 ID	815788206

## 四、Venus 平台上模型训练和预测流程



上述任务的地址: <http://venus.oa.com/#/project/program/6032706/workflow/6082674>  
Venus 中作业 ID: 815741326

### 4.1 训练节点

训练节点配置三个选项:

- 1) 运行文件或包: 上传模型代码源文件;
- 2) 训练样本路径: 指定为太极平台产出的 train 目录;
- 3) 训练样本数据周期: 指定模型训练的周期;

源代码见第六章。

### 4.2 模型预测流程配置

模型预测流程见我之前写的 wiki:

<https://iwiki.woa.com/pages/viewpage.action?pageId=955104697>

## 五、上线小流量实验流程

小流量上线步骤详情见附件 1.

<https://iwiki.woa.com/pages/viewpage.action?pageId=1564446452>

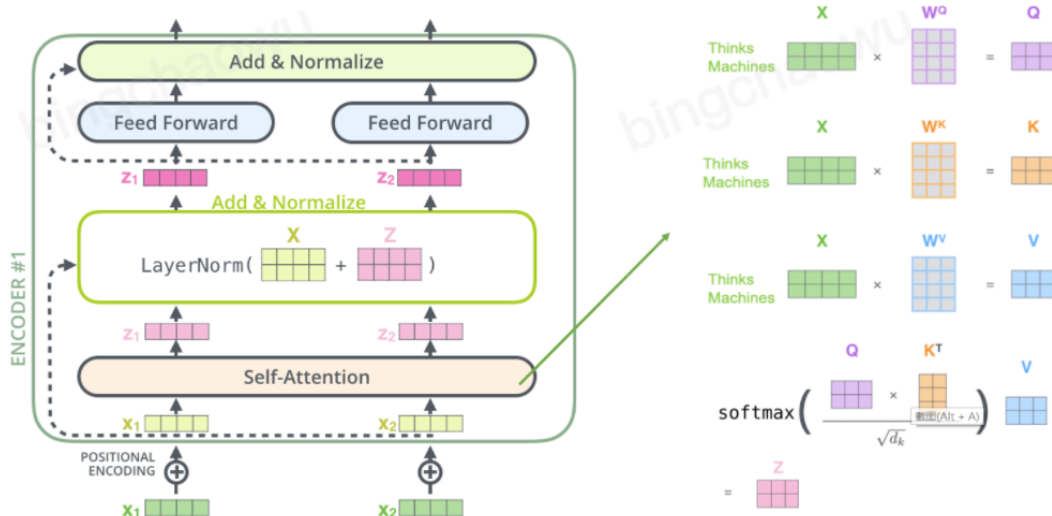
## 六、序列推荐模型优化

### 6.1 SASRec 模型(baseline)

Self-Attentive Sequential Recommendation (SASRec) 是基于 Transformer 的序列模型。Transformer 不同于 rnn-based 的序列模型，可以捕获长距离依赖关系，通过利用了 self-attention 机制可以在预测下一个 item 时候，更关注用户相关的历史行为数据，减小数据噪声，提高预测准确度。SASRec 应用于序列推荐领域能捕捉解决用户随时间变化的、动态的偏好，通过 self-attention 策略直接建模任意两个 items 之间的相似度，由此从用户的历史行为中寻找哪些 item 是"相关的"，并基于它们来预测下一个 item。

该模型与 Transformer 的编码层大体上是一致的，Transformer 中抛弃了传统的 CNN 和 RNN，整个网络结构完全是由 Attention 机制组成。核心思想是给定 target 中某个元素 Q, 通过计算 QK 的相似度得到每个 K 对应 Value 的权重系数，然后对 Value 进行加权求和。

如下图所示，模型主要由 self-attention 层和 feed forward 层构成；首先引入 positional embedding，将 concat 过 positional\_emb 的 seq\_emb 送入 self-attention 层，该层的核心公式为图中右下角的 softmax 公式，另外文章提出 t 时刻不能看到 t 时刻之后的信息，因此加上了 sequence mask，sequence mask 一般是通过生成一个下三角矩阵来实现的，上三角区域对应要 mask 的部分。self-attention 能够用自适应权重聚集之前所有 item 的 embedding，最终它仍然是个线性模型。为了增加非线性同时考虑不同隐式维度之间的交互，用了一个两层的 point-wise 前馈网络：作者使用卷积核大小为 1 的一维卷积实现的 PointWiseFeedForward 层。



SASRec 模型的目标函数采用二元交叉熵损失；seq\_embedding 经过 self\_attention 后得到 feature；feature 与 pos\_embedding 相乘结果越大越好，feature 与

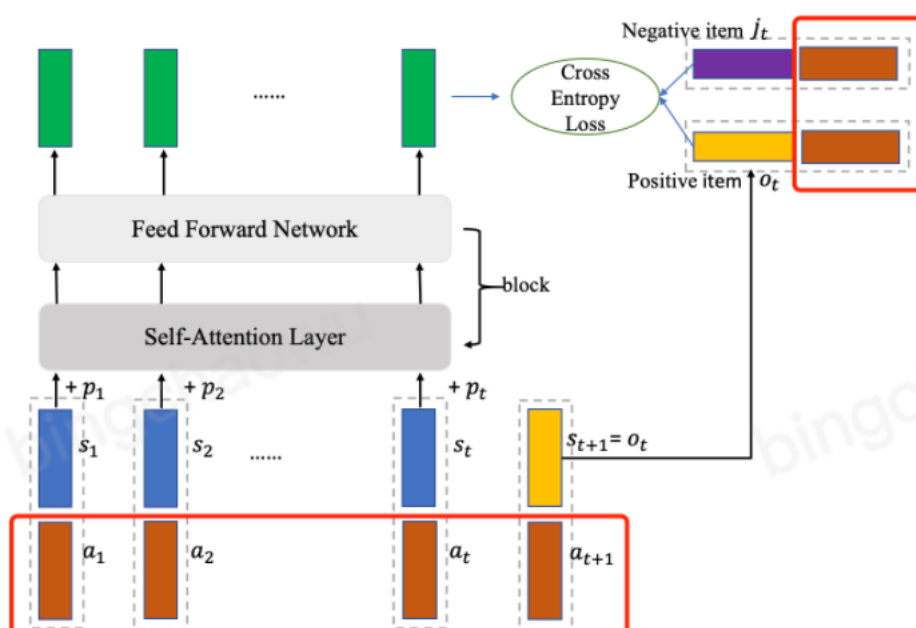
neg\_embedding 相乘结果越小越好。

$$-\sum_{S^u \in \mathcal{S}} \sum_{t \in [1, 2, \dots, n]} \left[ \log(\sigma(r_{o_t, t})) + \sum_{j \notin S^u} \log(1 - \sigma(r_{j, t})) \right]$$

源代码地址:	<a href="https://git.woa.com/jenesislui/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_baseline.py">https://git.woa.com/jenesislui/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_baseline.py</a>
上线效果	小流量验证时长显著正向
Venus 画布任务 ID	815639480

## 6.2 SASRec 模型融合 sideinformation

Side information 是用户或者 item 的辅助信息，例如用户的属性，物品的属性，物品的描述等，融合 side information 已经在学术界和工业界被广泛验证有效，能更加充分表达物品的语义信息。在本次实验中，我们选取的融合到 SASRec 的 side information 是**物品的二级分类**，具体策略是首先将物品的二级分类通过 embedding layer 转化为向量，之后分别拼接到输入的 item 表征和目前 item 表征，具体的模型如下：



分别在原始 SASRec 模型的输入层和目标 target 上拼接上物品的二级分类表征，由此让模型学习到语义知识更加充分的物品和用户表征，由此来提升推荐的准确性。

源代码地址:	<a href="https://git.woa.com/jenesislui/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_sideinfo.py">https://git.woa.com/jenesislui/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_sideinfo.py</a>
上线效果	小流量验证时长显著正向

Venus 画布任务 ID	815639480
---------------	-----------

### 6.3 FSDA 模型

Sasrec 模型融合 side information 是隐性考虑物品属性之间的关联，缺乏对属性之间的直接建模，而 FSDA[1]模型是通过分别单独建模物品 ID 序列和属性序列后，再将两者表征融合来构建损失函数，整体的模型框架图如下：

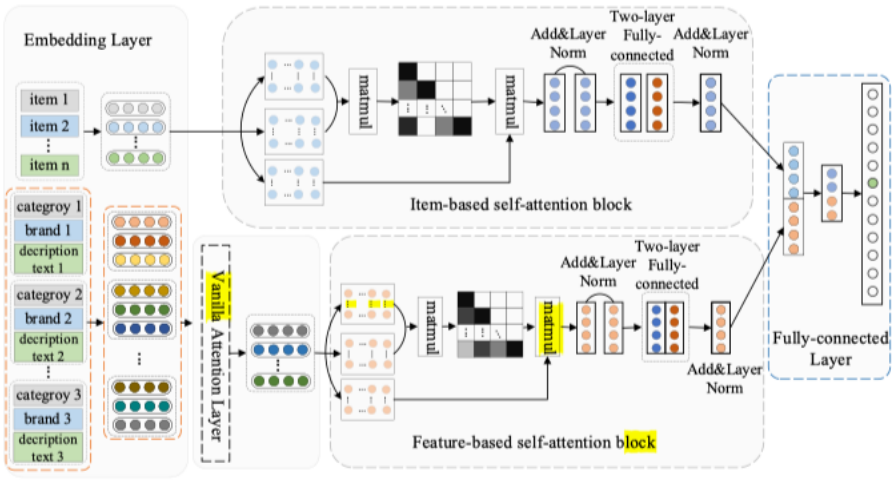


Figure 1: The Network Architecture of FSDA.

源代码地址:	<a href="https://git.woa.com/jenesisliu/recall_seq/blob/master/sasrec_u2i_bingchaowu/fsda.py">https://git.woa.com/jenesisliu/recall_seq/blob/master/sasrec_u2i_bingchaowu/fsda.py</a>
上线效果	未进行小流量实验
Venus 画布任务 ID	815908239

#### 参考文献

[1] Feature-level Deeper Self-Attention Network for Sequential Recommendation [Ijcai 2019]

### 6.4 SASRec 模型融合 hard 负样本

6.1, 6.2 和 6.3 节在建模优化目标时，只利用随机负样本，忽略了 hard 负样本，而 hard 负样本则能加精确建模用户的不感兴趣，因此本节将用户不感兴趣的快划负样本引入到序列推荐模型。(目前线上全量的模型版本)

详情见 wiki 地址:

<https://wiki.woa.com/pages/viewpage.action?pageId=1275801133>

源代码地址:	<a href="https://git.woa.com/jenesisliu/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_hard_neg.py">https://git.woa.com/jenesisliu/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_hard_neg.py</a>
上线效果	小流量验证时长显著正向
Venus 画布任务 ID	815741326

## 6.5 SASRec 模型融合 inbatch 内负样本

目前线上全量模型的负样本由随机负样本和 Hard 负样本构成，如何增多负样本数量是本节考虑的重点，现有 DSSM 等模型考虑 inbatch 内的其他样本作为当前样本的负样本，本节将 inbatch 内除了当前行为序列之外的其他序列中的样本作为负样本。随机负样本和 inbatch 内负样本可分别理解为基于均匀分布的采样和基于流行度分布采样的负样本，从而更充分建模用户的兴趣表征。但是目前上线的效果不佳，可能是三者负样本的数量比例未调整，后续调整比例可能会有一定的收益。

源代码地址:	<a href="https://git.woa.com/jenesisliu/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_inbatch.py">https://git.woa.com/jenesisliu/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_inbatch.py</a>
上线效果	小流量验证时长负显著
Venus 画布任务 ID	815812721

## 6.6 SASRec 模型融合 crossBatch 内负样本

除了在考虑 inbatch 内负样本之前，还考虑在不同 batch 内的行为序列中的物品作为当前样本的负样本。融合的策略与 inbatch 内负样本一致。

源代码地址:	<a href="https://git.woa.com/jenesisliu/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_cross_batch.py">https://git.woa.com/jenesisliu/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_cross_batch.py</a>
上线效果	小流量验证时长负显著
Venus 画布任务 ID	815823555

## 6.7 SASRec 模型融合 Next N items 建模

之前的版本均考虑 Next one item 构建模型的优化目标，受强化学习模型和 contrastive predictive coding (CPC) 的启发，从预测下一个视频扩展为预测未来 N 个视频的方式，在计算 loss 时考虑未来 N 个 target 视频带来的损失。

详细的 wiki 见:

<https://wiki.woa.com/pages/viewpage.action?pageId=955104704>



(1) 版本 1: 未考虑 hard 负样本的效果, 忘记添加 hard 负样本;

源代码地址:	<a href="https://git.woa.com/jenesisliu/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_next_n_items.py">https://git.woa.com/jenesisliu/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_next_n_items.py</a>
上线效果	小流量验证时长持平
Venus 画布任务 ID	815898402

(2) 版本 2: 考虑 hard 负样本的效果;

源代码地址:	<a href="https://git.woa.com/jenesisliu/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_next_n_items.py">https://git.woa.com/jenesisliu/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_next_n_items.py</a>
上线效果	未上小流量实验
Venus 画布任务 ID	815908253

## 6.8 SASRec 模型融合 Target Item 侧对比损失函数

为缓解 SASRec 模型在建模中对于稀疏物品学习不充分的问题, 参考文献[1]采样对比学习来学习到更加鲁邦的物品表征。具体是对目标物品进行采样得到多个正样本对, 再构建对比损失函数来作为推荐任务的辅助优化目标, 对比学习的形式化表示如下:

$$\ell_{\text{MINCE}} = - \sum_i \left[ \log \frac{\sum_{* \in \mathcal{P}_i} e^{(\hat{\mathbf{z}}_i^\top \cdot \mathbf{z}_*) / \tau}}{\sum_{* \in \mathcal{P}_i} e^{(\hat{\mathbf{z}}_i^\top \cdot \mathbf{z}_*) / \tau} + \sum_{* \in \mathcal{N}_i} e^{(\hat{\mathbf{z}}_i^\top \cdot \mathbf{z}_*) / \tau}} \right]$$

其中  $\mathbf{z}^*$  是 target item,  $\hat{\mathbf{z}}$  是经过 dropout 层后的表征, 具体的模型参考论文 [1].

源代码地址:	<a href="https://git.woa.com/jenesisliu/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_dropout_target.py">https://git.woa.com/jenesisliu/recall_seq/blob/master/sasrec_u2i_bingchaowu/sasrec_dropout_target.py</a>
上线效果	小流量验证时长持平
Venus 画布任务 ID	815822563

参考文献:

[1] Memory Augmented Multi-Instance Contrastive Predictive Coding for Sequential Recommendation [ICDM 2021]

## 6.9 基于 SASRec 的全序列模型

基于 SASRec 模型的全序列模型核心优化在样本侧的优化, 详情见 3.2 节, 模型侧的优化暂未进行。难点在于随着样本数量的不断增多, 导致模型的参数量剧增, 如何根据阈值找到合适的样本量是待解决的问题。



源代码地址:	暂时与线上全量模型一致
上线效果	小流量验证时长负显著
Venus 画布任务 ID	815788591

## 6.10 融合知识图谱(KG)的序列推荐模型

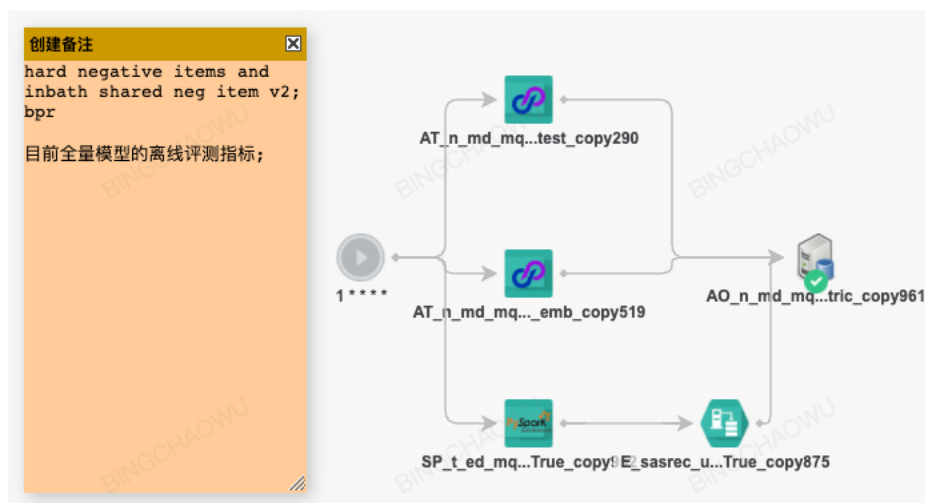
本节的工作是将知识图谱(KG)中蕴含的外部知识融合到序列模型，参考论文[1]通过构建基于用户行为的全局行为图，分别在全局图和单个用户行为序列上编码得到的语义表征，然后融合后来预测模型下一个物品。目前上述工作暂未开展，期待能有好的效果。

源代码地址:	<a href="https://anonymous.4open.science/r/CribiaNet-B3B8">https://anonymous.4open.science/r/CribiaNet-B3B8</a>
上线效果	未上线
Venus 画布任务 ID	只在公开数据集上验证效果

参考文献:

[1] Edge-Enhanced Global Disentangled Graph Neural Network for Sequential Recommendation [ICDM 2022]

## 七、Venus 上计算评价指标流程



离线评估流程的详细介绍见 wiki:

<https://iwiki.woa.com/pages/viewpage.action?pageId=1120561677>

本节介绍下每个节点的功能:

- (1) \*copy290: 产出 T+0 天的 item 池子;
- (2) \*copy519: 产出 T+1 天的用户表征;

- (3) \*copy875: 产出 T+1 天的 ground truth, 用户行为序列中的 last item;
- (4) \*copy961: 计算离线评估的节点;

Venus 作业 ID: 815797352

## 八、后续优化点

后续优化点分为两部分:

- (1) Wiki 上总结部分后续优化点:

<https://iwiki.woa.com/pages/viewpage.action?pageId=1150519195>

- (2) 序列推荐论文总结: 详情见附件 2:

<https://iwiki.woa.com/pages/viewpage.action?pageId=1564446452>