

目录

实习任务.....	2
报告内容.....	2
1 论文调研总结.....	2
2 论文复现.....	3
3 改进尝试.....	4
4 待解决问题.....	7
实习收获.....	7
附录 A.....	7
周记 1.....	7
周记 2.....	9
周记 3.....	11
周记 4.....	13
附录 B.....	15

实习任务

1 调研推荐系统冷启动近期研究工作（自 2017 年起，发表在 NIPS/ICML/ICLR/KDD/WSDM/IJCAI/AAAI 等重要国际会议的论文或者 survey），进行论文采集，并进行适当的归纳和整理；

2 选定一篇论文（其代码开源），复现论文中的实验结果；

3 对上述现有方法尝试进行改进，并通过实验验证改进后的效果；完成生产实习报告并在 GITHUB 上提交代码；

报告内容

1 论文调研总结

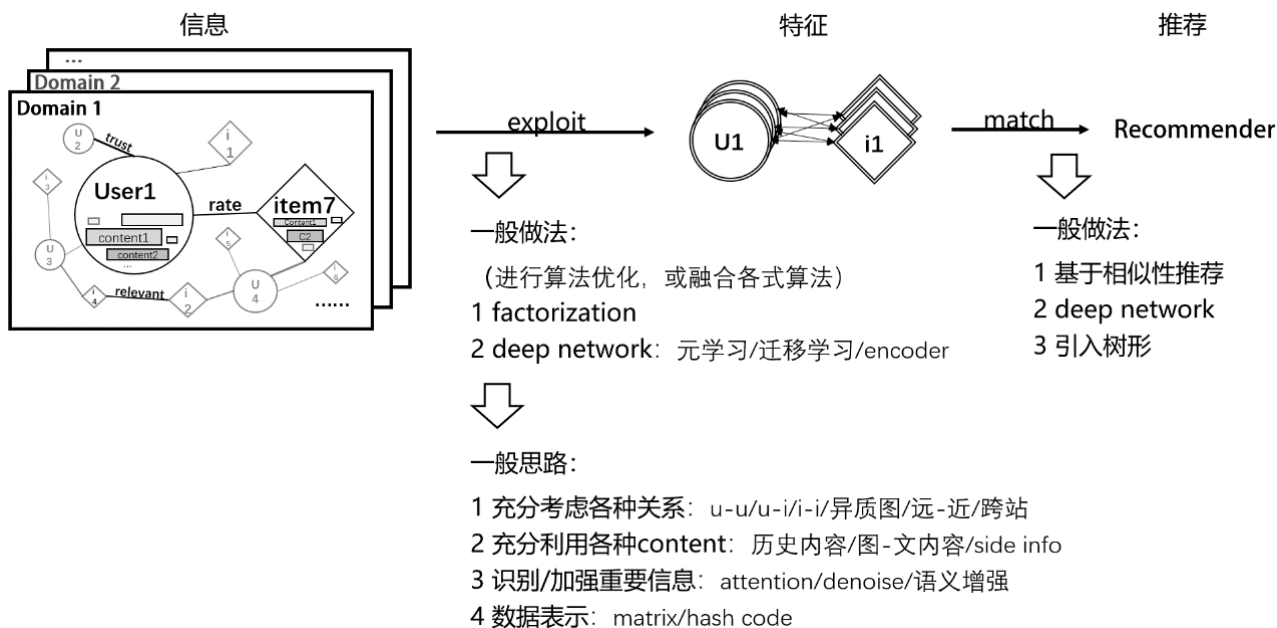
1.1 采集的论文见附录 B

1.2 论文阅读总结

1.2.1 说明

1) 本次共收集论文 29 篇，阅读了其中 6 篇，周记内有相关记录；2) 主要阅读论文 related work 和 conclusion 部分，了解论文提出的思路；

1.2.2 在阅读了解后，总结出冷启动研究思路如下：



2 论文复现

2.1 说明

1) 所选择论文为 KDD2020 的 MetaHIN1, 因为其内容较新且算法细节详尽;
 2) 主要工作: 仿照源码, 改进实现了论文的元学习模型; 因为考虑到论文核心非数据处理, 所以对于源码中数据处理和评估的部分, 我依循源码、只做了理解和修改适配; 3) 具体代码和电子版报告见 github²;

2.2 论文理解时的关键点

2.2.1 多层循环结构

本论文是根据 MAML 元学习架构的, 主要循环包括: 一、循环遍历每个 batch; 二、循环遍历每个 batch 中的每个任务 (注, 每个任务中包含若干 (item, rate), 它们同属于一个 user, 且分为支撑集和询问集);

在此之上, 本论文提出, 利用异质图元路径来增强语义。由于一个 user 在图中可处于多种元路径中, 所以还有一层循环来遍历每种元路径; 遍历完成后要整合来作为该 user 的计算数据;

2.2.2 参数及其更新

在 MAML 架构里, 参数有全局和局部两种状态; 全局参数一般等到一个 batch 结束之后再更新, 本论文源码将参数注册进模型, 在优化时会自动更新; 而局部参数是用来评估单个 batch 中所有用户的学习情况的, 经过局部学习之后, 看其

¹ Yuanfu Lu, Yuan Fang, Chuan Shi, 2020, Meta-learning on Heterogeneous Information Networks for Cold-start Recommendation. On KDD.

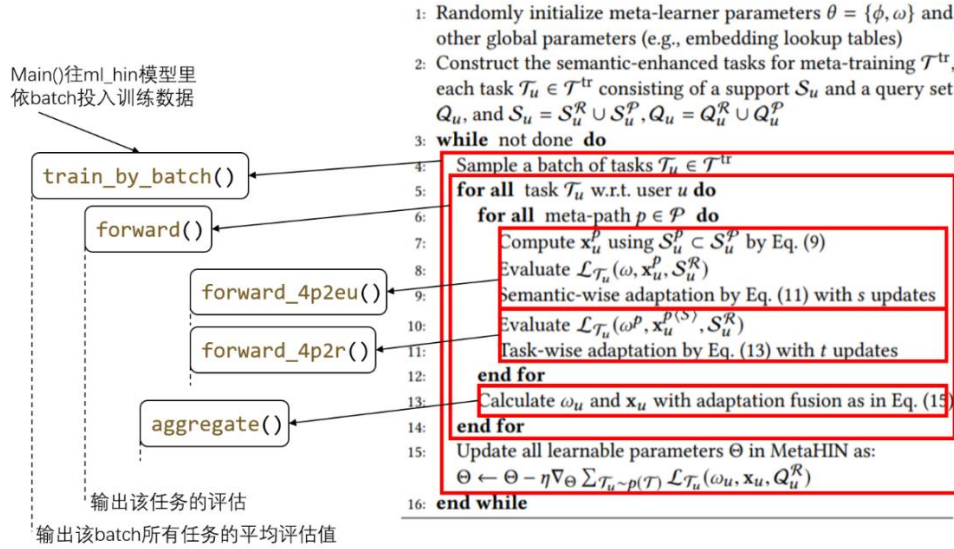
² <https://github.com/subin4git/MetaHIN>

预测效果，进而再以此更新全局参数；注意，局部学习时会有协同适应的环节，该处还有一层循环；

2.3 复现工作的细节

2.3.1 代码里模型函数的结构

Meta learning model 的重要函数与论文算法的关系如图：



其中，forward_4p2eu()和 forward_4p2r()函数负责局部地学习两个参数，其使用的 base learning 的模型分别为 class lp2eu 和 lp2r（取“learn parameters to embed user & rate”之意）；

注意两处实现：1)mean pooling 直接使用 torch.mean();2)user_embedding 通过 repeat()方法来升维，进而才与 item_embedding 连接，做感知机的输入；

2.3.2 参数选择和数据集取舍

实验只进行简单复现，故而1)未进行超参数调节的工作，直接使用论文推荐的参数值来看效果；除了 batch_size，将其设为 16 以满足内存限制；2)数据集只选择了 movielens、且舍弃了其中 user-movie-user-movie 的元路径数据，以满足内存限制；

2.4 实现效果

即，模型可以正常工作，其训练后 evaluating 的结果见表 1；

由于缺失一条元路径的关系，三个评估参数都没有原论文好，故而只与修改的模型进行比较；

3 改进尝试

3.1 源码矛盾处的改进

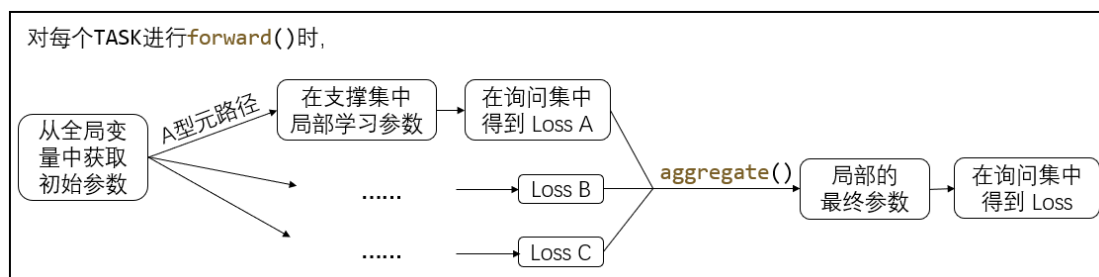
3.1.1 问题

表 1 模型评估效果

Scenario	MAE↓			RMSE↓			nDCG@5↑		
	模型 1	模型 2	模型 3	模型 1	模型 2	模型 3	模型 1	模型 2	模型 3
UC	0.9196	0.8996	0.9046	1.0797	1.0816	1.0799	0.7923	0.7903	0.7868
IC	0.9563	0.9545	0.9573	1.1359	1.1314	1.1364	0.7666	0.7657	0.7632
UIC	0.9584	0.9579	0.9610	1.1526	1.1510	1.1507	0.7808	0.7805	0.7718
NC	0.9196	0.9208	0.9176	1.1042	1.1078	1.1001	0.8119	0.8142	0.8063

*UC: user cold-start; IC: item cold-start; UIC: user-item cold-start; NC: non cold-start;

模型 1: 基本的模型; 模型 2: 经过方案 1 修正的模型; 模型 3: 经过方案 2 引入记忆的模型;



阅读论文源码，发现一处矛盾的地方：

论文愿意是，在“一个任务”下，我们需要考虑“多种元路径”所能带来的语义增强，如图：

其中，“在询问集中得到 Loss A”的步骤，进行训练时如此，是为了根据各个元路径的影响力来调整参数；可是在进行测试时，询问集在训练局部参数时，应该是未知的，所以这一步骤应该有所不同；而论文源码中，测试和训练时同用一个 forward()，因此在逻辑上有矛盾；

3.1.2 改进方案

每个元路径各自维护一个变量；

训练每个任务时，这些变量记录对应元路径在询问集中得到的 loss、并与之前的 loss 求平均；而在测试时，我们只需要使用这些变量来调整得到最终参数即可；

3.1.3 效果及讨论

训练后 evaluating 的结果见表 1；比较模型 1 和模型 2，可以看到：

(1) 基本上，模型 1 与模型 2 的评估值差别不大，即，修正对训练效果没有显著影响；此时，使用修正后的模型会更符合逻辑；

(2) 在 item 非 cold-start 的场景中 (UC 和 NC)，模型 1 和 2 的 MAE 有明显差别；这说明，在 item 信息丰富的情况下，每个 user 的不同元路径会有不同的比例；

3.2 引入记忆矩阵

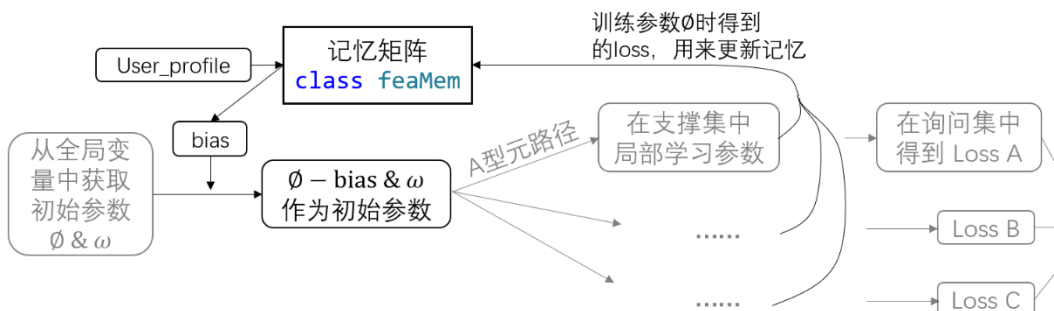
3.2.1 背景

KDD2020 的另一篇冷启动论文³提出了记忆增强的元学习优化算法，其中有一个核心观点是，引入记忆矩阵，使得初始参数进一步与 user 相关；

3.2.2 引入方案

(1) MAMO 模型的内容丰富，在此我仅截取其中的“用户特征记忆模型”来改进当前复现的论文；相关逻辑如图：

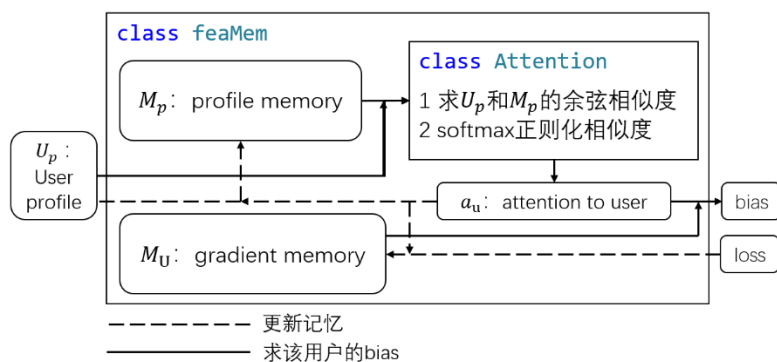
对每个TASK进行forward()时，



改进的思路解释：

原先，参数学习路径是：所有用户共有的参数 \rightarrow 用户自己的参数；加入记忆矩阵后，学习的路径就变为：所有用户共有的参数 \rightarrow 用户所属类型/所属圈子的参数 \rightarrow 用户自己的参数；

(2) 具体 feaMem 模型如图：



关于代码实现：

我基本上沿用了 MAMO 论文的源代码，主要只做了适应性的修改；

3.2.3 效果及讨论

训练后 evaluating 的结果见表 1；比较模型 1 和模型 3，可以看到：

(1) 对于 MAE 和 RMSE 评估值，模型 3 没有显著的提升；

(2) 而对于 nDCG@5 评估值，模型 3 相对模型 1 有稳定的改良（IC 场景除外）；这可以解释为：记忆化之后的模型 3 加强了对 user 特征的定位，从而能更好地利用 user 的信息，但是，还依旧对 item 信息缺少的情况无能为力；

³ Manqing Dong, Feng Yuan, Lina Yao, Xiwei Xu, Liming Zhu, 2020, MAMO: Memory-Augmented Meta-Optimization for Cold-start Recommendation. On KDD.

但必须要说明的是，引入记忆矩阵后，一次 forward 的时间代价就被大大增加；并且 MAMO 原论文中提出的，“使评估值在各个场景下变得稳定”的效果，也没有在本次实验中体现出来，这有待考究；

4 待解决问题

(1) 学习和阅读中还有很多的疑问有待解决，如周记中的“？”标注、尤其周记 3 中关于论文算法的疑惑；

(2) 完整复现论文还需要有“超参数学习”、“模型比较”两个环节；原论文的数据非常好，是否是偶然还有待完整的验证；

(3) 改进方案还可以再琢磨琢磨；

(4) 对深度学习的更高层的总结、代码还需规范化；

实习收获

(1) 第一次较真地去了解深度学习的细节；

(2) 通过研读论文了解了推荐系统冷启动；

(3) 粗浅地了解了会议论文的基本内容/方法；

(4) pytorch 等相关代码的编写能力得到提升；

附录 A

周记 1

本周主要工作：

阅读学习“花书”《深度学习》（阅读了机器&深度学习基础部分）。

学习总结如下：

1 深度学习发展历史

（待制图）

收获：人工神经网络和其实和神经科学有很大不同，研究深度学习，更多还是在算法、模型的设计研究上；

2 样本参数估计与贝叶斯方法

基于 Frequency 的学习	基于 Bayesian 的学习
$(X, Y)_{train} \rightarrow \text{learn } \theta \text{ for } f(x; \theta)$	$p(\theta) \text{ as prior} \rightarrow p(\theta x_1 \dots x_m)$
X 为随机变量	把 θ 作为随机变量
用点值 θ 来做预测	用 θ 的全分布来做预测（积分方式可防过拟合） $p(x_{n+1} x_1 \dots x_n) = \int p(x_{n+1} \theta) \cdot p(\theta x_1 \dots x_n)$

θ 的不确定性体现于方差	θ 的不确定性体现于 <i>prior</i>
	1 在数据少时泛化更好；多时计算困难； 2 用概率反应“知识状态的确定程度”；

3 为什么相中最大似然？

最大似然 ($\arg \max_{\theta} (\sum \log(p) \cdot 1/m)$) 用于使模型分布靠近训练集数据分布；

原因 1：不同于一般方法 ($\arg \max_{\theta} (\prod p)$)，使用求和而非乘积，避免计算问题；

原因 2：用 KL 散度衡量分布之间的差别时，公式中的负对数可于最大似然中的对数对应；KL 散度非负，计算时亦方便；

4 关于交叉熵

MSE 为模型分布和高斯模型分布之间的交叉熵；

许多输出单元都会有 e^x ，会在 x 趋向负无穷大时失去梯度，此时可用交叉熵；如，用于 Multinoulli 输出分布的 softmax 单元，加上对数：

$$\log \text{softmax}(z) = z_i - \log \sum \exp(z_j)$$

则此时，当 z_i 趋向最大化时，惩罚项 $\log \sum \exp(z_j) \rightarrow \max(z_j) = z_i$ ；

有一种特殊的学习，是学习 x 与 y 的统计量之间的参数；有两种方法可以使用，一种为变分法，该方法使用均方误差和平均绝对误差，不利于梯度下降（与饱和的函数组合，会产生小梯度）；另一种即为使用交叉熵；

5 神经网络中考虑用线性单元做二元输出

$$p(y = 1|x) = \max\{0, \min\{1, \omega^T x + b\}\}$$

则此时无法进行梯度下降；

6 无监督与有监督

其实就是有没有 y 值的区别，无监督学习的是 x 的概率分布；

7 性能度量 MSE、MAE 和 RMSE

MSE 和 MAE 比较参考：<https://www.jianshu.com/p/1ff7ae7ea9ef>

MSE/RootSE	赋予异常点更大的权重；	异常情况重要时用
MAE	使用固定学习率时，梯度难收敛；	异常只是受损数据时用

8 最快梯度下降的核心

在 x 多维、且无法推导参数正规公式时，我们从一个点不停下降来找最优解；此时，与该点梯度反向的方向下降速度是最快的。

9 有关激活函数

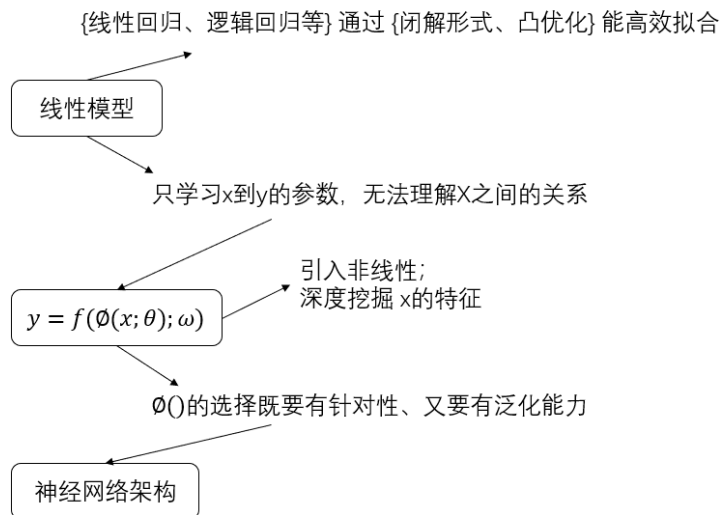
参考博客 <https://zhuanlan.zhihu.com/p/73214810>

激活函数要考虑的问题：一、梯度爆炸；二、梯度消失；三、是否 0 均值；四、Dead ReLU 问题；前两个随神经网络加深而影响计算、第三个则会改变初始分布；第四个是因为负值全变 0、导致部分神经元失效，可以通过 Xavier 初始化+调节 lr 来改善；

常见激活函数有：

饱和 (左右趋于 0)	Sigmoid	平滑	适合概率值处理
	Tanh	平滑+zero-centered	
非饱和	Relu	快	适合深层网络 (更好保留原信息)
	LeakyRelu/PRelu	快+解决“dead”问题	
	ELU	更快+避免梯度消失	

10 为什么提出神经网络：从线性到非线性



11 正则化的含义

即在损失函数中加一个惩罚项，如

$$J(\omega) = MSE_{train} + \alpha \omega^T \omega$$

则此时，参数训练会向 L2 范数小的地方前进；

12 数据的使用

一般分为三个部分

训练集	训练部分	训练模型
	验证集	可以估计泛化误差，从而训练超参数
测试集		估计最终的泛化误差

13 当测试集不够大，误差小时，可以使用 k-折交叉验证；核心思想是，从数据中随机采样来验证；代价是：不存在平均误差的方差的无偏估计；

周记 2

本周主要工作：

粗浅了解冷启动推荐，收集相关论文并初步阅读。

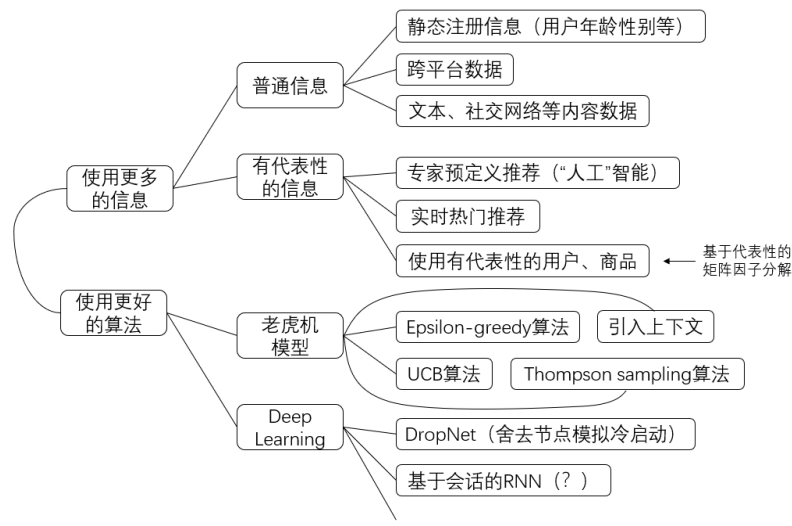
学习总结如下：

1 冷启动种类

给新用户推荐旧物品、把新物品推送给老用户、新的系统中给新用户推荐新物品；

2 冷启动解决方案

总结基于知乎问答：<https://www.zhihu.com/question/19843390/answers/updated>



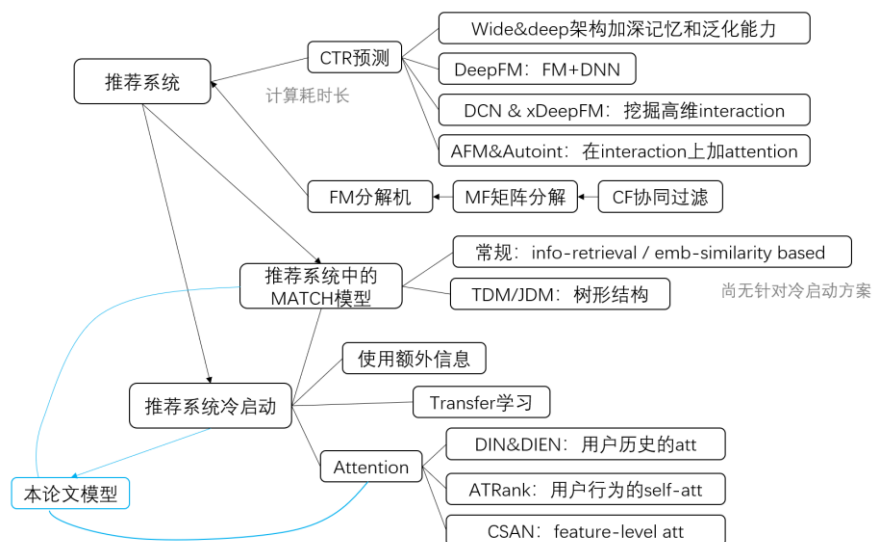
3 领域相关关键词

冷启动 (cold-start)、稀疏 (sparse)、低秩 (low-rank)、推荐 (recomm);

相关领域：数据挖掘、人工智能/深度学习/神经网络；

相关会议：除了rank靠前的顶会，由博客得知，还有 RecSys 等专业性强的推荐系统会议 (https://blog.csdn.net/qq_27032425/article/details/75099826)；

4 论文阅读之 Internal and Contextual Attention Network



个人理解：

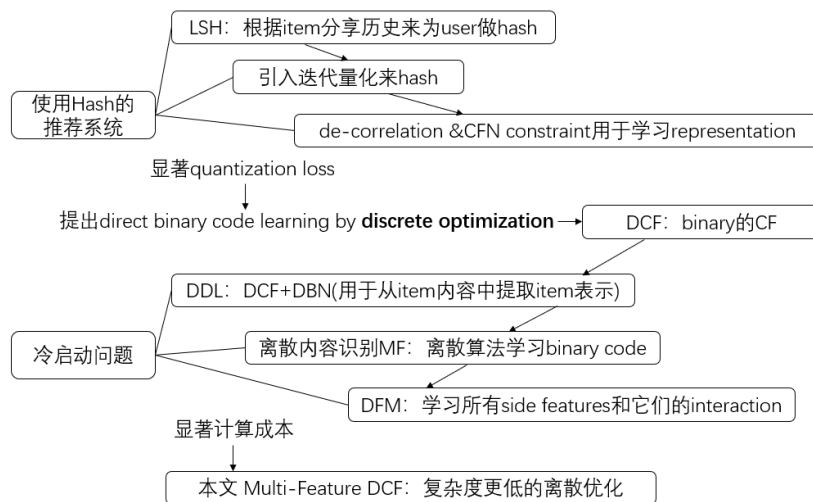
本文主要是进行 CTR 预测；本文中的 attention 是这么用的：一、每个 channel 的所有用户的历史行为，构成这个 channel 的特征、且每个 channel 都有不同 fields；二、文本 attention

层对每个 channel 的 fields 进行,然后所有 channel 的 fields 对应整合;三、内部 attention 层对每个整合了的 fields 进行;四、两层 attention 抓取所有用户(?)的行为作为 emb 并进行 match;

结论:

一、冷启动 channel 受非冷启动 channel 影响显著;二、分开考虑 feature fields 是有必要的;

5 论文阅读之 Multi-Feature Discrete Collaborative Filtering



个人理解:

一、使用 hash code +协同过滤来做推荐系统;二、提出了更快的离散优化算法;算法核心是 ALM 增广拉格朗日乘子,使在约束下更快收敛;并且直接的目标函数避免了中间矩阵的计算(?);文章结论部分有对论文自身优点的归纳;

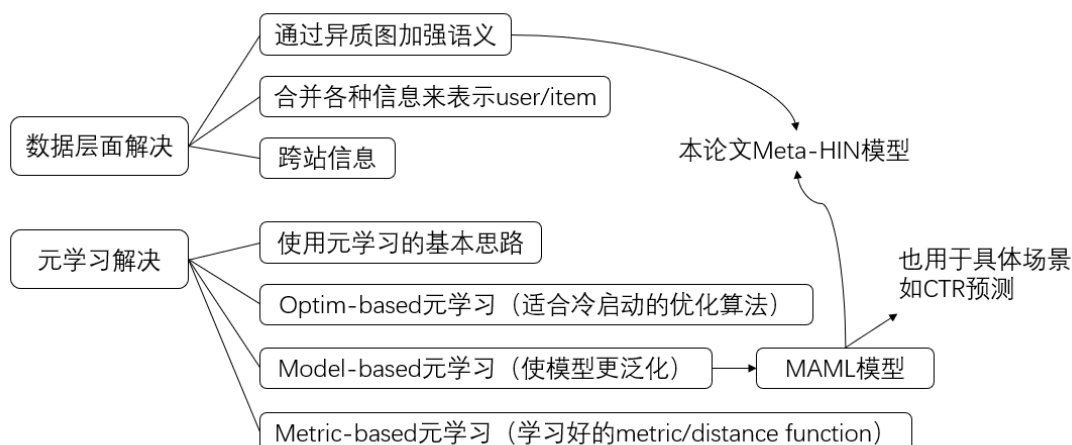
周记 3

本周主要工作:

尝试复现了论文的部分内容。

学习总结如下:

1 论文阅读: 相关研究



元学习的好处：能在 few examples 上更快地学习新的 task；
MAML 在这篇论文之前，已经应用于冷启动，且效果很好；

2 论文阅读：公式探究

2.1 user_emb 公式

$$\mathbf{x}_u = g_\phi(u, C_u) = \sigma(\text{MEAN}(\{\mathbf{W}\mathbf{e}_j + \mathbf{b} : j \in C_u\})), \quad (6)$$

在这里，我们用与 user 相关的所有 item_emb 来进行 embed；分两步，一、由于商品数量多，属高维，需要经过 mean pooling，即平均池化，通俗的讲就是降维表示（关于池化种类以及作用可参考 <https://mp.weixin.qq.com/s/ISvHyUrXpxGTCMVib-ptnw> 或《深度学习》9.3）；二、激活，引入非线性；

论文在这里使用 LeakyRelu 做激活，猜想原因为，一、使用 Relu 为普遍做法；二、LeakyRelu 解决“dead”问题且公式简单；

2.2 评分公式

$$\hat{r}_{ui} = h_\omega(\mathbf{x}_u, \mathbf{e}_i) = \text{MLP}(\mathbf{x}_u \oplus \mathbf{e}_i), \quad (7)$$

在这里，我们输入（用户，物品），输入评分；两个输入进行简单连接、并使用最基本的三层感知机；

2.3 把全局参数转化进当前元路径的空间（Transformation Function）

$$\omega^p = \omega \odot \kappa(\mathbf{x}_u^{p(S)}), \quad (12)$$

$$\kappa(\mathbf{x}_u^{p(S)}) = \text{sigmoid}(\mathbf{W}_\kappa \mathbf{x}_u^{p(S)} + \mathbf{b}_\kappa), \quad (18)$$

想法就是，通过全局参数与 user_emb 进行 element-wise product 来实现；element-wise product 要求两个乘数矩阵的维度相同，所以需要转换函数；

在这里，我们通过 Linear 直接转换维度并激活；注意两点，一、维度转换时使用的参数是可以学习的；二、为什么使用 sigmoid 函数？

查看论文源代码时，注意到代码通过 nn.Linear 来实现，每次转化维度时，都初始化其参数，并没有体现出“学习”的过程，为什么呢？

2.4 几种元路径聚合时

$$\omega_u = \sum_{p \in \mathcal{P}} a_p \omega_u^p, \quad \mathbf{x}_u = \sum_{p \in \mathcal{P}} a_p \mathbf{x}_u^{p(Q)}, \quad (15)$$

where $a_p = \text{softmax}(-\mathcal{L}_{\mathcal{T}_u}(\omega_u^p, \mathbf{x}_u^{p(Q)}, Q_u^R))$ is the weight of the p -

在这里, 我们根据对应元路径的 loss 来判断该路径数据的权重; 注意两点, 一、使用 softmax 是为了归一化; 二、为什么在这里要先对 loss 取负值再归一? 简单计算可以发现, 该操作使得取值大的 loss 占比没那么小, 取值小的 loss 占比没那么小;

3 论文阅读: 模型分析

3.1 元学习的效果

比较 MetaHIN-BM(只有 base-model)、MetaHIN-FT(BM 基础上进行 fine-tune) 和 MetaHIN, 发现一、有没有 FT 其实区分不大, 这可能是冷启动限制的; 二、元学习提升效果明显, 是 prior 的功劳;

3.2 语义增强和协同适应的效果

比较 MeLU(只是使用异质信息作为 features)、MetaHIN-TA(只进行 task-wise 参数的学习, 无语义适应)、MetaHIN-ID(进行简单的语义适应, 但不全局学习更新) 和 MetaHIN, 发现一、语义增强是有效果的; 二全局参数学习 general 的知识也是有效果的;

4 论文阅读: Meta-HIN 的比较对象

传统推荐模型	FM	用了许多额外信息
	NeuMF	矩阵分解+MLP(提高低维特征), 一种先进的 NN
	GC-MC	使用 GCN(图特征提取)来 embedding
考虑异质图的模型	mp2vec	在图上 Random walks 来学习 embedding
	HERec	矩阵分解来提取图中信息
经典冷启动模型	DropoutNet	NN
	MetaEmb	用于 CTR 预测的元学习
	MeLU	MAML 框架的元学习

6 fast-weight: 局部参数用词; fine-tune: 一种模型微调的规范 (?);

7 协同适应/协同过滤

即利用相同兴趣圈的其他人来为目标做推荐;

8“异质图”在论文中的含义: 即比普通 user-movie 多考虑了 actor、director 等异质点及连接它们的异质线;

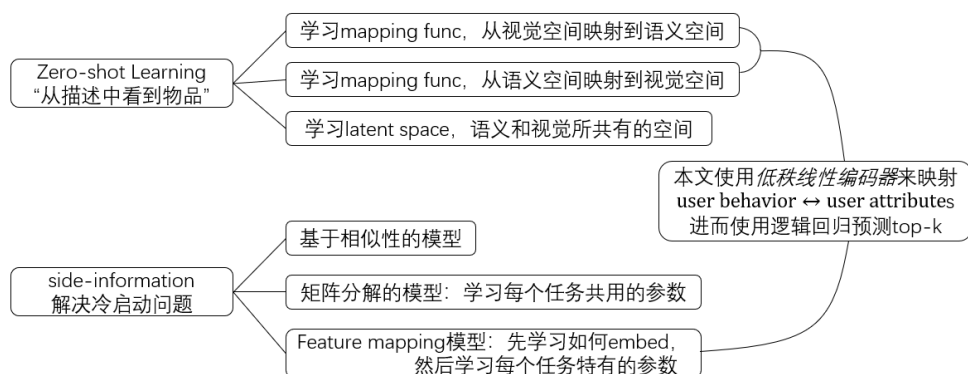
周记 4

本周主要工作:

阅读相关论文。

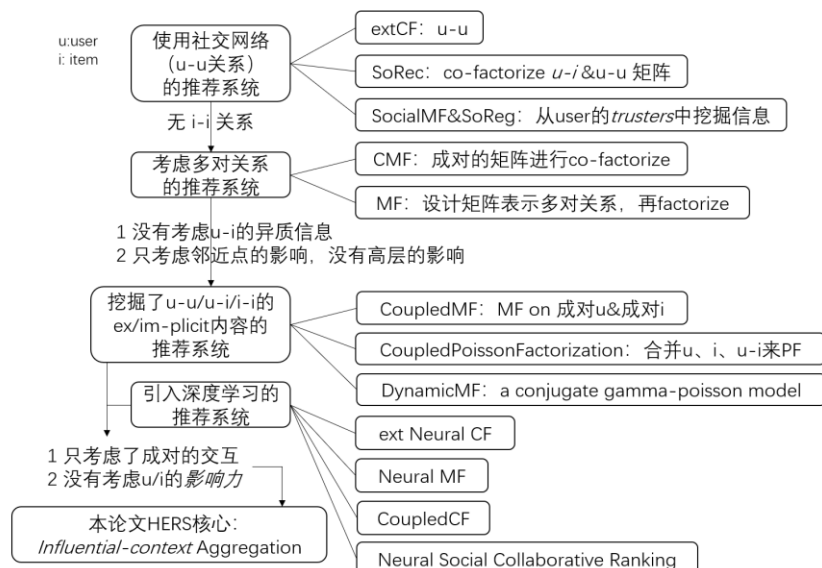
学习总结如下:

1 论文阅读之 From Zero-Shot Learning to Cold-Start



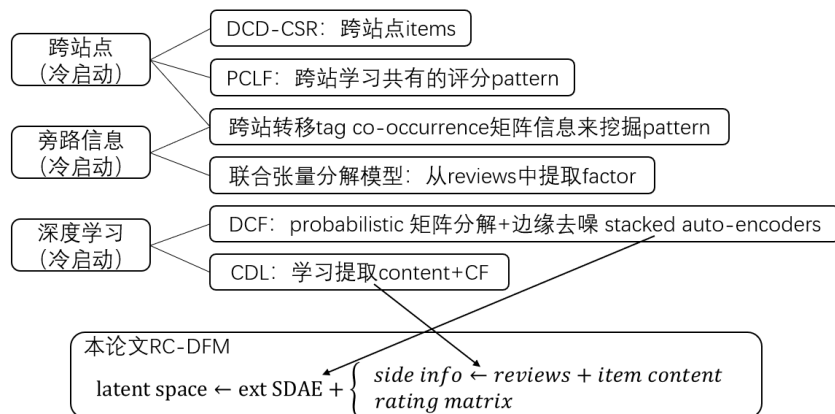
个人理解: 从 side info 中挖掘用户信息, 应当是个基本的想法, 为何本论文出众? 猜测是编码器的运用部分有特别的细节;

2 论文阅读之: Modeling Influential Contexts with Heterogeneous Relations



个人理解: 提出了“大 V 用户/热门 item”对图的影响, 充分利用这个影响来挖掘推荐的方向; 除了邻近点, 还要考虑远点的思路;

3 论文阅读之 Deeply Fusing Reviews and Contents



个人理解：融合了丰富文字信息的学习，又因为信息丰富所以需要去噪；

附录 B

收集论文列表

1 筛选条件

17 年至今，在每个会议的官网、或者 dblp 对应 accepted paper 收录页面（页面中仅包含论文标题及作者）检索关键词“cold”，内容相关即收集；特别说明的是，发现推荐系统本身论文众多，冷启动只是其中的一小部分；

2 检索的会议

NIPS、ICML、ICLR、KDD、WSDM、IJCAI、AAAI、SigIR、CIKM、PAKDD、Recsys；

3 列表内容

注：检索不到相关论文的会议不列出

3.1 NIPS (Neural Information Processing Systems)

2017
<i>A Meta-Learning Perspective on Cold-Start Recommendations for Items</i>
<i>DropoutNet: Addressing Cold Start in Recommender Systems</i>
<i>Cold-Start Reinforcement Learning with Softmax Policy Gradient</i>

3.2 KDD (Knowledge Discovery and Data Mining)

2020
<i>MAMO: Memory-Augmented Meta-Optimization for Cold-start Recommendation</i>
<i>Meta-learning on Heterogeneous Information Networks for Cold-start Recommendation</i>
2019
<i>MeLU: Meta-Learned User Preference Estimator for Cold-Start Recommendation</i>

3.3 IJCAI (International Joint Conferences on Artificial Intelligence)

2020
<i>Internal and Contextual Attention Network for Cold-start Multi-channel Matching in Recommendation</i>
2019
<i>Cold-Start Aware Deep Memory Network for Multi-Entity Aspect-Based (冷启动识别)</i>
2017
<i>Recommendation vs Sentiment Analysis: A Text-Driven Latent Factor Model for Rating Prediction with Cold-Start Awareness</i>

3.4 AAAI (Association for the Advancement of Artificial Intelligence)

2020
<i>Multi-Feature Discrete Collaborative Filtering for Fast Cold-start Recommendation</i>
2019

<i>Deeply Fusing Reviews and Contents for Cold Start Users in Cross-Domain Recommendation System</i>
<i>From Zero-Shot Learning to Cold-Start Recommendation</i>
<i>HERS: Modeling Influential Contexts with Heterogeneous Relations for Sparse and Coldstart Recommendation</i>
2017
<i>Low-rank Linear Cold-Start Recommendation from Social Data</i>

3.5 SigIR(Special Interest Group on Information Retrieval)

2020 (China)
<i>AR-CF: Augmenting Virtual Users and Items in Collaborative Filtering for Addressing Cold-Start Problems</i>
<i>CATN: Cross-Domain Recommendation for Cold-Start Users via Aspect Transfer Network</i>
<i>Content-aware Neural Hashing for Cold-start Recommendation</i>
<i>Recommending Podcasts for Cold-Start Users Based on Music Listening and Taste</i>
<i>DCDIR: A Deep Cross-Domain Recommendation System for Cold Start Users in Insurance Domain</i>
<i>Joint Training Capsule Network for Cold Start Recommendation</i>
<i>A Heterogeneous Graph Neural Model for Cold-start Recommendation.</i>
<i>A Heterogeneous Information Network based Cross Domain Insurance Recommendation System for Cold Start Users</i>
2019
<i>Warm Up Cold-start Advertisements: Improving CTR Predictions via Learning to Learn ID Embeddings</i>

3.6 CIKM(Conference on Information and Knowledge Management)

2019
<i>What You Look Matters?: Offline Evaluation of Advertising Creatives for Cold-start Problem</i>
<i>Semi-Supervised Learning for Cross-Domain Recommendation to Cold-Start Users</i>
2018
<i>Attention-based Adaptive Model to Unify Warm and Cold Starts Recommendation</i>

3.7 PAKDD(亚太 KDD)

2017
<i>Integrating Reviews into Personalized Ranking for Cold Start Recommendation</i>

3.8 Recsys

2019
<i>CB2CF: A Neural Multiview Content-to-Collaborative Filtering Model for Completely Cold Item Recommendations</i>
<i>Domain Adaptation in Display Advertising: An Application for Partner Cold-Start</i>