

# 基于子图推理的可解释推荐

邓晓峰

2021214317

2022 年 6 月 23 日

## 1 介绍

基于知识图谱的推荐系统的研究近些年发展迅速，主要是把知识图谱作为辅助信息与推荐系统结合，利用知识图谱中的信息挖掘用户和物品之间深层次的关系，可以提升推荐效果。相比黑盒模型，在知识图谱上采用推理模型，可以产生带有解释性的预测结果，同时也可能提升预测结果的准确性。

如今基于知识图谱的推荐的推荐方法通常分为两类，基于嵌入的方法 (embedding based methods) 和基于路径的方法 (path-based methods)。基于嵌入的方法主要是通过知识图谱的边或者结构信息，学习实体的表征，对实体进行编码，但缺乏可解释性。基于路径的方法，试图构建一个解释路径，得到结果的同时也得到知识图谱中路径作为解释，但问题是它无法构建路径以外的结构化信息来更好地解释预测结果。为此，我们在知识图谱上进行子图推理，目的是学习一个动态的子图，该子图以用户节点开始，预测的产品尾节点结束。

图推理需要学习实体、关系和它们的组成规则来使用结构化的知识并产生结构化的解释。这部分的工作我们使用图神经网络 (GNNs) 实现。具体地，我们递归地聚合来自邻居节点的信息，用于更新节点信息。经过  $T$  次迭代后，每个节点都包含来自它的  $T$  跳邻居节点的结构信息。

在使用图神经网络学习图的结构信息后，我们需要使用注意力机制得到推荐结构和推荐的解释。为此我们使用了新的图注意力机制，以产生子图用于得到推荐商品并做出结构化的解释。我们的注意力机制主要考虑三点，(1) 对于给定的一个子图，首先我们在其内部根据注意力分数选择几个节点，然后主要关注这些节点的邻居，以进行下一步的子图扩展；(2) 我们

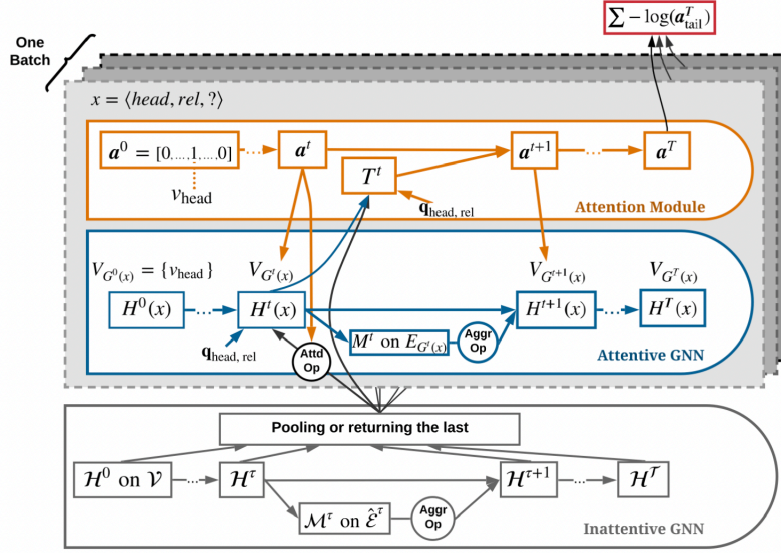


图 1: 模型架构

在模型的各个步骤中传播注意力，以捕捉长期的依赖性；(3) 我们的注意力机制通过级联计算模拟推理过程，并且该计算与节点表征计算分离并在节点表征计算之上执行；

模型把图中实体分为两个状态，一个是注意力机制前的无意识状态，注意力机制后的有意识状态。我们使用两个图神经网络对这两种状态进行编码。我们构建了依赖输入的子图为了有效地执行消息传递，并在完整图上执行完整的消息传递，以获得子图约束之外的局部的特征。我们在两个图神经网络之间应用注意力机制。其中全局图神经网络在执行注意力机制之前运行，称为无注意力图神经网络 (Inattentive GNN, IGNN)，而子图的 GNNs 则在每个注意力机制后产生的子图上运行。称为注意力图神经网络 (Attentive GNN, AGNN)。IGNN 为 AGNN 提供在完整图上计算的表示。AGNN 进一步处理子图中相关节点的表示，以产生更契合推荐任务的语义。

## 2 模型

### 2.1 模型的架构设计

我们的模型架构如图一所示：

IGNN 模块：执行完全的消息传递，以计算全图的节点表表征。

AGNN 模块：在修剪过的子图上执行消息传递，以计算与输入有关的节点表表征，它也利用了来自 IGNN 的基础表征。

注意力模块：执行一个流式注意力转换过程，以 IGNN 和 AGNN 的节点表表征为条件，它的输出只影响 AGNN，并且在最后会将注意力分数最高的节点作为最终模型的输出。

## 2.2 IGNN 模块

我们使用标准的图神经网络来实现 IGNN 模块，如果完整图上有很多的边，我们会采用对图中的边进行采样，只会在采样的边的上计算节点表征。对于执行模型的同一批输入，IGNN 生成的节点表征会在不同的任务中共享，所以，IGNN 的复杂度不会随着同一批次中的输入大小影响。IGNN 的计算如下：

信息编码： $\mathcal{M}_{<\acute{v},r,v>,;}^T = \varphi_{IGNN}(\mathcal{H}_{v,;,e_r,\mathcal{H}_{v,;}}^T)$ , where  $<\acute{v},r,v> \in \hat{\mathcal{E}}^T$

信息聚合： $\bar{M}_{v,;}^T = \frac{1}{NT(v)^{\frac{1}{2}}} \sum_{\acute{v},r} \mathcal{M}_{<\acute{v},r,v>,;}^T$ , where  $<\acute{v},r,v> \in \hat{\mathcal{E}}^T$

状态更新： $\mathcal{H}_{v,;}^{T+1} = \mathcal{H}_{v,;}^T + \delta_{IGNN}(\mathcal{H}_{v,;}^T, \bar{M}_{v,;}^T, e_v)$ , where  $v \in V$

$\hat{\mathcal{E}}^T$  表示每一步的采样边集合， $\varphi_{IGNN}$  和  $\delta_{IGNN}$  使用两层全连接神经网络 (MLP)， $v$  表示节点， $r$  表示关系， $e_r$  和  $e_v$  分别表示关系和节点的初始嵌入向量。

IGNN 在运行了  $T$  个步骤后，我们输出一个结果集合，用  $H = H_T$  表示，输入下游模块。

## 2.3 AGNN 模块

AGNN 是依赖输入，这意味着它的节点状态取决于输入用户。我们在子图上运行 AGNN，每个子图都以输入用户为条件。使用  $H_{v,;}^t$  表示节点  $v$  在子图中的表征，当  $t=0$  时，我们从用户节点 head 开始， $V_{G^0(x)} = \{v_{head}\}$ ，AGNN 的计算如下：

信息编码： $\mathcal{M}_{<\acute{v},r,v>,;}^T = \varphi_{AGNN}(\mathcal{H}_{v,;,c_r,\mathcal{H}_{v,;}}^T)$ , where  $<\acute{v},r,v> \in E_{G^t(x)}$ , and  $c_r[e_r, e_{head}, e_{rel}]$  代表背景向量

信息聚合： $\bar{M}_{v,;}^T = \frac{1}{NT(v)^{\frac{1}{2}}} \sum_{\acute{v},r} \mathcal{M}_{<\acute{v},r,v>,;}^T$ , where  $<\acute{v},r,v> \in E_{G^t(x)}$

状态伴随函数： $\tilde{H}_{v,;}^{t+1}(x) = a_v^{t+1} W \mathcal{H}_v$ ,  $a_v^{t+1}$  是节点的注意力分数

状态更新:  $H_{v,:}^{T+1} = H_{v,:}^T + \delta_{AGNN}(\mathcal{H}_{v,:}^T, \bar{M}_{v,:}^T, c_v^{t+1}), c_v^{t+1}[\tilde{H}_{v,:}^{t+1}(x), e_{head}, e_{rel}]$   
也代表背景向量

其中  $e_{rel}$  为用户与商品的关系, 由具体任务进行选择, 我们使用一个节点状态伴随函数来传递节点表示信息, 从 IGNN 传递给 AGNN, 并由标量注意力分数  $a_v^{t+1}$  加权。并由一个可学习矩阵  $W$  进行映射。在进行 AGNN 阶段, 对于节点在  $V_{G^0(x)}$ , 我们会对各个节点使用 IGNN 输出的结果进行初始化, 而不在  $V_{G^0(x)}$  的节点保持 0 状态。

## 2.4 Attention 模块

在  $T$  步中, 注意力由一连串的概率分布表示, 使用  $a^t(1, 2, 3, 4, \dots, T)$ 。其中初始分布  $a^0$  是一个独热向量  $a^0[v_{head}] = 1$ , 为了传播注意力, 在每一步都需要计算过渡矩阵  $T^t$ 。该注意力机制基于 IGNN 和 AGNN, 我们计算两种类型的交互在  $\acute{v}$  和  $v$  之间:  $H_{v,:}^T$  表示在 AGNN 中的表征,  $\mathcal{H}_{v,:}^T$  表示在 IGNN 中的表征。Attention 的计算如下:

$$T_{:,v}^t = softmax_{v \in N^t(v_t)} \left( \sum_r a1(H_{\acute{v},:}^t(x), c_r(x), H_{v,:}^t) + a2(H_{\acute{v},:}^t(x), c_r(x), \mathcal{H}_{v,:}^t) \right)$$

$$a1(.) = MLP(H_{\acute{v},:}^t(x), c_r(x))W_1MLP(H_{v,:}^t, c_r(x))$$

$$a2(.) = MLP(H_{\acute{v},:}^t(x), c_r(x))W_2MLP(\mathcal{H}_{v,:}^t, c_r(x))$$

$W_1$  和  $W_2$  是两个可学习的矩阵, 而 MLP 是一个一层的神经网络。由于节点是由上一轮注意力值大小修剪而来, 可能会发生注意力值的损失, 从而减少注意力值总和, 所以我们使用重新规范化的注意力,  $a^{t+1} = T^t a^t / \|T^t a^t\|$ 。

## 3 总结

该模型主要参考的是 Xiaoran Xu 在 ICLR2020 上发表的论文《DYNAMICALLY PRUNED MESSAGE PASSING NETWORKS FOR LARGE-SCALE KNOWLEDGE GRAPH REASONING》, 虽然该模型的效果在推荐上比某些推荐模型的效果要好, 但应该也仍有提升空间。而该模型比较大的问题在于模型耗费的时间的一般的推荐模型要多, 这也是后面的研究需要改进的地方。