一、KGAT: Knowledge Graph Attention Network for

Recommendation（知识图谱注意网络推荐）

1．论文信息简介：

（1）原文地址：https://doi.org/10.1145/3292500.3330989

（2）内容介绍：基于知识图谱+图注意力网络的推荐系统（KG+GAT）

（3）发布信息：

2019年8月4日至8日，在美国阿拉斯加州安克雷奇举行的KDD ' 19。

“KDD是由ACM主办的国际数据挖掘领域最顶级会议,同时被CCF(中国计算机学会)列为A类会议,本年度注册参会人数超过3400名。”

（4）作者信息：

Xiang Wang、National University of Singapore（新加坡国立大学）

Xiangnan He、University of Science and Technology of China（中国科学技术大学）

Yixin Cao、National University of Singapore（新加坡国立大学）

Meng Liu、Shandong University（山东大学）

Tat-Seng Chua、National University of Singapore（新加坡国立大学）

（5）学习前的准备

了解推荐系统基础概念、协同过滤推荐算法、Trans系列（TransE、TransH、TransR）、softmax函数等

（6）讲述内容

传统推荐算法的缺陷、协同知识图谱CKG、KGAT模型（嵌入层、注意力嵌入传播层

、预测层）

2.论文内容

（一）传统推荐算法的缺陷

图表, 图示

描述已自动生成

图1 A toy example of collaborative knowledge graph

在电影场景下，关系类型简介：

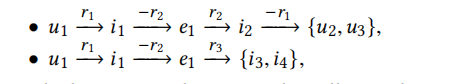
r1：观看过、r2：导演过、r3：指导拍摄或参与过、r4是——。

如上图：用户u1 看了电影i1 ，这个电影是 ei 导演的。

传统的CF（协同过滤）方法会着重去找那些也看了电影i1 的用户，比如u4、u5 。

SL（监督学习）方法会重点关注那些有相同属性e1的电影，比如 i2。

很显然这两类信息都可以作为推荐信息的补充，但是KGAT之前的模型不能做到上面两者信息的融合，而且这里的高阶关系也可以作为推荐信息的补充的。比如图中黄色框图里的用户u2和u3看了同样由 e1 导演的电影 i2 , 还有灰色框图里电影i3、i4有e2的指导拍摄[1]。

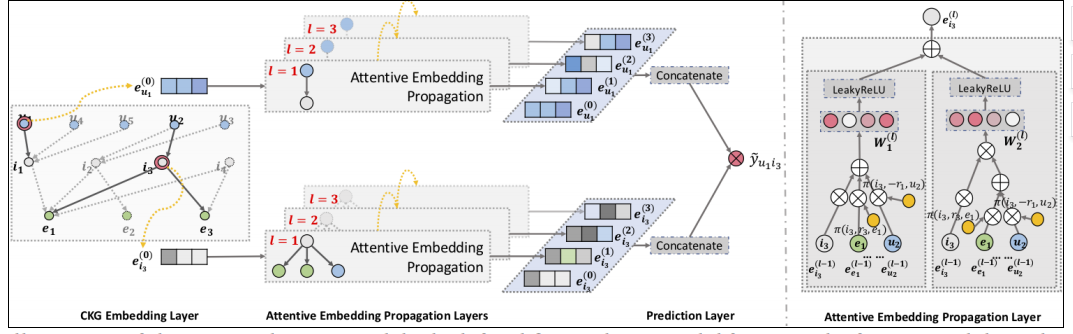


(二)协同知识图谱CKG

为了解决上面提到的问题，作者提出 collaborative knowledge graph (CKG)方法，将图谱关系信息及用户user点击商品item的交互图融合到一个图空间里。这样就可以融合CF信息及KG信息，同时也可以通过CKG发现高阶的关系信息[2]

用户物品二部图：用户物品交互图，有交互的用边相连。  
知识图谱：包含物品属性描述的图。  
协同知识图谱：这里将二部图和物品实体知识图谱结合成一张图。

（三）模型介绍



整个模型可以分为3层，分别是嵌入层、注意力嵌入传播层和模型预测。

输入：由G1（用户物品交互图）和G2（知识图谱）衔接得来的CKG；

输出：一个预测U、I之间交互概率的预测函数。

（1）Embedding Layer嵌入层

简要来说就是在CKG上用TransR模型（知识图谱嵌入方法）将实体和关系参数化为向量表示。

输入：CKG

输出：实体和关系参数化为向量表示(eui(0) and eti(0)  and ri)

（2）Attentive Embedding Propagation Layers注意力嵌入传播层

先看单层：

1. 信息传播（information propagation）
2. 知识意识注意（knowledge-aware attention）
3. 信息聚合（information aggregation）
4. 信息传播

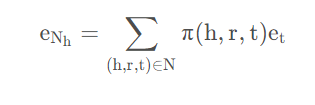
一个实体可以出现在多个三元组里面，如：

文本

描述已自动生成

i2可以靠e1、e2两个邻居的属性细化自身嵌入，从而递归的细化u2的嵌入。

模拟消息传播过程，现给定一个实体h，Nh = {(h,r,t)|(h,r,t)∈G}是其作为头实体的三元组，为了刻画h的一阶连通性结构，计算Nh的线性组合：



这里的eNh可以理解为h一阶邻居信息的聚合。

其中Π(h,r,t)控制着边(h,r,t)上每次传播衰减因子，表示有多少信息从t传播到h。

输入：实体h的一阶邻居、Π（h，r，t）

输出：h一阶邻居信息的聚合（一个向量）

1. 知识意识注意

通过注意力机制来计算Π(h,r,t)

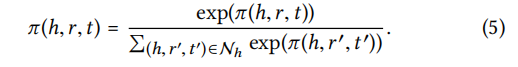


选择tanh（）作为非线性激活函数。

这使得注意力得分依赖于关系r空间中eh和et之间的距离，t越靠近h，Π越大。

作者只在这些表示上使用内积，并将注意力模块的进一步探索留作以后的工作。（可进一步改进）

之后，通过采用softmax函数（归一化指数函数）对与h相连的所有三元组的系数进行规范化：



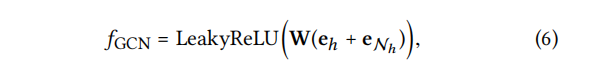
最后的注意得分能够提示哪些邻居节点应该给予更多的权重，也能够作为建议的解释。

1. 信息聚合

三种，分别是GCN聚合器、GraphSage聚合器、双交互聚合器。

GCN Aggregator：

GCN聚合器将两个表示相加并应用非线性变换。

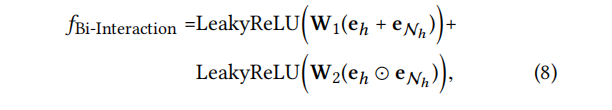


GraphSage Aggregator：  
GraphSage聚合器拼接两个表示，然后是非线性转换：



Bi-Interaction Aggregator：

双交互聚合器（作者提出），考虑eh和eNh之间的两种特征交互：



W1，W2∈Rd′×d为可训练权重矩阵，⊙表示元素乘积。

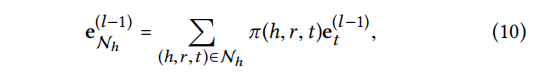
与GCN和GraphSage聚合器不同，双交互聚合器对eh和eNh之间的特征交互进行了编码,这使得传播的信息对eh和eNh之间的关联性敏感。（更好）

再看多层：

可以进一步堆叠更多的传播层来探索高阶连接性信息，收集从更高跳邻居传播的信息。更正式地说，我们递归地将实体的表示形式表示为：



其中，在l-1自我网络中为实体h传播的信息定义：



et（l-1）是从先前的信息传播步骤生成的实体t的表示，它存储来自其（l-1）跳邻居的信息；eh（0）集作为初始信息传播迭代的eh。它进一步有助于实体h在层l上的表示。因此，在嵌入传播过程中，可以捕获高阶连接。高阶嵌入传播可以将基于属性的协作信号无缝地注入到表示学习的过程中[4]。

（3）Model Prediction预测层

在执行L层之后，我们获得了用户节点u的多层表示，即{eu（1），··，eu（L）}；类似于项目节点i，{ei（1），···，ei（L）}。由于第l层的输出是[图1](#_top)所示根在u（或i）处的l层树结构深度的消息聚合，因此不同层的输出强调不同阶的连接信息。因此，我们采用层聚合机制将每一步的表示连接成一个向量：

图片包含 文本

描述已自动生成

其中||是拼接操作，通过这样做，我们不仅可以通过执行嵌入传播操作来丰富初始嵌入，还可以通过调整L来控制传播强度。

最后，我们对用户和项目表示进行内积，以预测它们的匹配得分：

图片包含 文本

描述已自动生成

**参考文献**

[1]版权声明：本文为CSDN博主「小爷毛毛（卓寿杰）」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：https://blog.csdn.net/u011239443/article/details/116941892

[2]版权声明：本文为CSDN博主「Angeliaaa」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：https://blog.csdn.net/Angeliaaa/article/details/123273427

[3]版权声明：本文为CSDN博主「VieleZMaya」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：<https://blog.csdn.net/VieleZMaya/article/details/128043290>

[4]版权声明：本文为CSDN博主「大数据机器学习实验室」的原创文章，遵循CC 4.0 BY-SA版权协议，转载请附上原文出处链接及本声明。

原文链接：https://blog.csdn.net/diaokui2312/article/details/107757560