# 一种双向注意力机制的知识图谱神经网络推荐算法

A **g**raph **n**eural network **r**ecommender algorithm based on **b**idirectional **a**ttention (BAGNR)

2021年4月13日星期二 蓝色修改 wxl

红色字体 建议删除或从新写

**摘要：**在推荐系统中，基于知识图谱的神经网络与传统神经网络相比，前者以图形作为输入，可以很好地将节点信息和拓扑结构相结合进行预测。然而现有的方法中，很少考虑图结构中的对称关系以及信息聚合时梯度消失的问题。为此，本文将图神经网络与对称注意力机制相结合采用双向翻译模型对知识图谱信息进行特征的嵌入表示，并将双向嵌入模型与注意力机制相结合，使得注意力机制在决策权重时考虑的关系更全面。其次，在对节点和邻居信息进行训练过程中，为避免引入过多的参数而导致计算量增加和过拟合问题，引入了动态激活函数以提高模型的健壮性。最后，在两个真实数据集上与经典算法进行对比，验证了本文所提的算法有效性。

**关键词**：双向嵌入，注意力机制，知识图谱，推荐

**1.引言**

在这个信息爆炸的时代，解决信息过载问题，推荐系统是非常成功的。最早提出来的协同过滤算法，利用用户-项目的交互信息来实现预测，是实现有效的个性化推荐的基本算法[1,2]。通常来说，协同过滤模型中有两个关键的组成部分：一是将用户和项目的交互信息转换为矢量表示；二是基于转化为矢量的嵌入信息构建历史交互。在众多协同过滤算法中，矩阵分解是基于潜在特征推荐的主流方法之一，它将用户和项目投影到一个共享的潜在空间中，利用潜在的特征向量来表示用户或者项目，采用内积的形式进行建模。与此相关的许多工作都致力于增强矩阵分解，例如基于邻居的模型相结合[4]，将其与项目内容的主题模型相结合[5]，并将其拓展到因子分解机[6]以实现特征的通用建模。尽管矩阵分解对于协同过滤是有效的，但矩阵分解性能会受到交互函数线性内积简单选择的影响。

为了解决矩阵分解内积简单选择的影响，He等人[2]提出了神经协同过滤模型，使用深层神经网络从数据中学习交互函数，采用非线性的内积来取缔线性操作。另外，基于翻译的CF模型使用欧几里德距离度量作为交互函数[3]。尽管这些方法是有效的，但在面对冷启动问题仍具有欠缺，并不能产生足以令人满意的CF嵌入。现有的方法只利用描述性特征来构建嵌入函数，而不考虑用户与用户之间潜在的关系，将每个交互作用独立的建模，不足以从中提取具体的协同信号。为此，文献[]提出将知识图与用户-项目交互的二部图进行结合构成协同知识图谱来解决用户或项目间的高阶关系,如图1所示。



图1 用户、项目与实体之间的高阶关系

图1 左图是用户项目的交互二部图，右图是协同知识图谱。左图中用户*u1*与项目*i2*交互，用户*u3*与项目*i2*交互，找到一条*u1*→*i2*­­→*u3*这条关系，通过挖掘用户与项目之间的反馈信息，用户*u3*的信息就会反馈给用户*u1*，这样能够找到*u1*→*i2*→*u3*→*i3*路径信息，随后将项目*i3*推荐给用户*u1*。而协同知识图谱中拥有丰富的语义信息，例如， *u1*→*i2*→*e2*这条高阶关系，通过挖掘*e2*反馈信息，最终得到*u1*→*i2*→*e2*→*i1*，最终将*i1*推荐给用户*u1*。

为了捕获协同知识图中的高阶隐式关系。有学者利用神经注意机制[8,9]来获取传播过程中每个邻居的权重，因此级联传播的注意力权重可以揭示高阶连通性的重要性。文献[KGAT]以端到端的方式显式地建模了知识图谱中关系的高阶连通性，但该方法嵌入实体的所有近邻实体的语义信息，导致计算量过于庞大。除此之外，在KGAT的注意力传播层中尾实体到头实体的传播信息是以关系为条件，而信息的传播是双向的，并没有体现信息的充分性。

本文的主要贡献如下：(1)在知识表示过程中，考虑到实体之间的双向性来对实体进行嵌入表示以提高实体的度量精度。(2)在矩阵分解的用户-项目关系的基础上，深层次挖掘用户-项目-用户、用户-项目-用户-项目等高阶关系，并结合注意力机制模型来揭示节点的高阶连通性。(3)为了避免梯度消失问题，采用多通道的激活函数针对不同的高阶关系进行修正，从而增加反馈的信息量。(4)在真实数据集上与经典算法比较，实验结果表明了本文所提出的算法有效性。

**2. 相关工作**

**2.1 协同过滤算法**

协同过滤算法是推荐系统传统的算法[11,12]。协同过滤算法将用户和项目交互信息矢量化作为嵌入层，并重新构造用户和项目的历史交互关系来学习嵌入参数。早期的协同过滤算法模型，例如矩阵分解[13],预测了用户（项目）的ID投射到嵌入的向量。近年基于神经网络的协同过滤——神经协同过滤[2]和LRML[14]，采用神经网络的方式学习用户与项目潜在的交互关系，并采用共享的嵌入层，来增强神经网络的交互建模。(非常的不顺，建议删除)

**2.2 注意力机制与推荐系统**

注意力机制起源于模仿人类的思维方式，后被广泛应用于机器翻译、情感分类以及机器视觉等。谷歌发表的《Attention is all you need》[参考文献]一文引起学术界广泛关注，相关学者专家在注意力机制投入大量精力。陈等人[参考文献]提出的注意力机制的协同过滤模型则是将注意力机制与协同过滤相结合，利用分层的神经网络，并根据项目级别的项目和组件级别的内容(不是特别能理解想表达的意思)对用户的偏好分数进行建模，解决了项目与组件级隐式反馈的问题。阿里巴巴深度兴趣网络DIN模型[参考文献]从用户兴趣多样性和局部信息的相关性出发，对每个用户的兴趣表示都赋予不同的权值，然后加权求和。 XX等人[参考文献，Next Item Recommendation with Self-Attention]提出在推荐系统中使用自注意力机制，把推荐问题当作一个序列的问题，并对历史交互中的物品的序列的相关性进行加权，来预测下一个可能发生交互的物品。

**2.3知识图谱与推荐系统**

知识图谱的火热影响了许多方面的学术工作。基于特征的推荐算法[参考文献]从知识图谱中抽取一些用户和物品的属性作为特征，而未考虑属性之间的关系信息。~~放到传统的模型中，但是只引入了实体特征，没有一如关系的特征~~。基于异构信息网络的推荐方法[参考文献]是将知识图谱视为一个异构信息网，然后构造物品之间的基于元路径或元图的特征，这类方法充分且直观地利用了知识图谱的网络结构，但手动审计的元路径或者元图在时间中很难以达到最优。XX~~去年最新~~提出一种基于RNN的推荐模型[参考文献Explainable Reasoning over Knowledge Graphs for Recommendation]，在知识图谱中对用户和物品之间存在的关联路径进行建模，并向用户提供可解释性的推荐。

**3.BAGNR模型**

本文模型主要由嵌入层、修正注意力传播层以及预测层组成。嵌入层，将CKG(中文表述)模型参数化，将每个节点参数化为一个向量；修正注意力传播层，从节点的邻域中进行嵌入以更新其表示后，再利用修正后的知识注意力机制来获取传播过程中每个邻居的权重；预测层，从所有传播层聚合用户和项目的表示，并输出预测的匹配分数，如图2所示。



图2 BAGNR模型框架

**3.1 嵌入层**

知识特征表示是将知识图谱中的实体和关系嵌入到低维稠密的向量空间中[19].通过对图中实体和关系的三元组结构进行嵌入表示,保留其在原始空间中拓扑结构,对于知识图谱的扩展及推理能力的提升具有重大意义.用一个三元组(头实体*h*,关系*r*,尾实体*t*)代表一条知识,例如三元组(中国首都,地点,北京)表示中国的首都是北京,对于图中每一个存在的三元组*(h,r,t)*,实体和关系采用基于翻译的表示方法[20]进行嵌入.如式(1)所示.

 (1)

其中,*er h*，*er t*为*eh，et*在关系空间*r*中投影向量, *eh，et∈*和*er∈*分别表示*h,r,t*的初始化向量.

**3.2 修正的注意力机制**

知识图谱中蕴含着丰富的语义信息，使用TransR将知识图谱中的语义信息向量化，将实体投影到关系空间中，~~通过注意力机制控制边~~*~~(h,r,t)~~*~~每次的衰减因子，因为采用的是注意力机制，这使得注意力的得分过分依赖与关系r空间的两个实体的距离。~~注意力机制公式如式(2)。

 (2)

其中*eh,et*是在关系*r*空间投影表示*，Wr∈Rk\*d*是关系*r*的转换矩阵(解释k和d)。

由于TransR模型只考虑了节点的单向关系，而生活中存在反馈信息的双向关系。例如，在图1协同知识图谱中，*u1*通过项目*i2*与实体*e2*交互，*u4*通过项目*i3*与实体*e2*交互，丰富自身特性(主语是谁？)，然后贡献给*u4*，同时*u4*贡献少量的信息给*u1*，基于这种机制在用户、实体及节点邻居之间进行的高阶信息传播，同时借鉴文献[KGAT]，本文提出了双向加性注意力机制，如公式(3)所示。

 (3)

其中，是权重参数。改进后的注意力机制不仅关注于从头实体尾实体这种单向的关系，还对从尾实体到头实体可能存在的双向关系进行了关注，使得注意力机制在决策权重的时候，考虑关系更全面。

最终采用softmax函数对于*h*相连的所有三元组进行归一化，如公式(4)所示。

 (4)

~~因此，修正后注意力得分更能够充分地提示哪些邻居节点应该给予更多的关注来捕获协作信号，向前传播过程中，注意力也会提示要关注的部分数据。~~

**3.3 信息聚合的优化**

~~贝叶斯个性化排名算法(BPR)作为成对序列模型,被广泛应用于推荐系统中以实现个性化排名.在隐式反馈中,它假设用户参与的项目排名应高于用户未参与的项目排名.即：对于给定三元组<~~*~~u~~*~~,~~*~~i,j~~*~~>,若用户~~*~~u~~*~~参与项目~~*~~i~~*~~而未参与项目~~*~~j~~*~~,则将该三元组定义为正反馈并标记为+1；否则定义为负反馈,标记为0.其目的为最大化用户参与的项目与未参与的项目之间的距离.BPR目标函数如式(5)所示.~~

~~ (5)~~

~~其中,表示用户~~*~~u~~*~~对项目~~*~~i~~*~~的偏好程度,~~*~~σ~~*~~(~~*~~x~~*~~)表示sigmoid函数,~~*~~λ~~*~~表示模型的正则化参数用来避免过拟合,并假定Θ的先验分布是正态分布.由于BPR中含有大量的训练实例,因此通常采用随机梯度下降(SGD)优化训练模型,获得训练参数,从而依据获得的参数计算值,对指定用户~~*~~u~~*~~产生个性化推荐列表.~~

信息聚合的激活函数的在反向转播中起着重要的作用。

信息聚合在最终模型的预测上得分起着重要的作用，其中激活函数的在反向转播中起着重要的作用。考虑到不同路径的高阶关系具有不同的权重，本文非线性*PRelu*作为激活函数，如公式（6）(公式需要从新排序)所示。

 (6)

其中，*i* 表示不同的通道数。对于第*i*个通道拥有一个*iβ*，并且这些*iβ*会随着数据进行修正，从而增加反馈的信息量。同时也避免了引入过多的参数导致的计算量增加和过拟合问题，考虑划分通道，共享一部分参数。本文考虑让每次聚合过程中线性变换的参数矩阵中每一列划为同一个通道。

**4 实验**

本实验编程语言采用Python，服务器配置Precision 7920 Tower(2颗Xeon 金牌 5118 (12C 2.3GHz)/128G内存)。

主要解决以下两个问题：

(RQ1) BGANR算法与经典算法在推荐性能方面进行比较。

(RQ2)寻找模型中超参α和β的最优值。

**4.1数据集**

为了评估本文所提出BGANR模型的性能, 在Amazon-book [25]和Last-FM[7] 的两个公开的数据集上进行验证，~~Last-FM是全球最大的音乐推荐数据集，采集时间戳从2015年1月到2015年6月的数据集的子集来作为我们实验的数据集。Amazon-book 是推荐系统中被广泛使用的数据集，~~为了保证数据集的质量，本文保留了用户和项目至少有过交互的数据信息，表1为数据集的统计信息

表1 数据集描述

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Amazon-book | Last-FM |
| User-Item Interaction | #user | 70679 | 23566 |
| #Item | 24915 | 48123 |
| #Interactions | 847733 | 3034796 |
| Density | 4.81E-04 | 2.67E-03 |
| Knowledge Graph | #Entities | 88572 | 58266 |
| #Relations | 39 | 9 |
| #Triplets | 2557746 | 464567 |

除了用户和项目之间的交互信息外，还需要为每个数据集构造项目知识图谱。对于*Amazon-Book*和*Last-FM*，我们采用[21 N-2020/10/5]等人的方法，如果存在映射关系，则通过标题匹配的方式将项目映射到自由实体。我们还考虑到与项目对齐的直接相关的三元组。

与目前大多数只提供单跳的感知数据集不同，还将包含双跳相邻的实体的三元组考虑在内。为了保证知识图谱的高质量，我们筛出了不常见或者交互信息量少的实体。(不通顺)

本文选择80%的交互历史信息作为训练集，其余作为测试集。同时采用了交叉K折交叉验证，从训练集中随机的选择10%的交互历史信息验证机(没有理解)。~~对于每次观察到的用户-项目交互信息，都视为一个积极的实例，并执行负采样的策略来匹配用户以前没有交互信息的负例。(~~建议删除)

**4.2 基准算法**

* BPRMF[参考文献]：该模型针对用户的隐式反馈信息解决推荐排序问题。
* NFM[参考文献]：结合深度神经网络和FM建模更高阶的特征之间的关系。
* CKE[参考文献]：结合协同过滤学习节点的潜在表示。
* CFKG[参考文献]：构造用户项目知识图并结合用户行为和物品信息进行推荐。
* KGAT[参考文献]：强调协同知识图谱中显式建模高阶关系信息,以用于个性化推荐。

**4.3 评价指标**

召回率如式(7)所示。

 (7)

准确率如式(8)所示，其中*U*是所有用户集合，*R(u)*为单个用户的推荐列表集合，*T(u)*表示推荐列表集合中用户感兴趣的集合。

 (8)

命中率如式(9)所示，其中*U*是所有用户集合，*hitu*为用户在推荐列表中位于测试集的商品个数，*testu*表示用户测试集商品个数。

 (9)

*NDCG*如式(10)所示。

 (10)

其中, *DCG*如式(11)所示,*reli*为单个相关性分数。

 (11)

**4.4 实验对比与分析**

**4.4.1 与基准经典算法性能比较**

Amazon-book，这里需要对图进行描述，需要一段话。

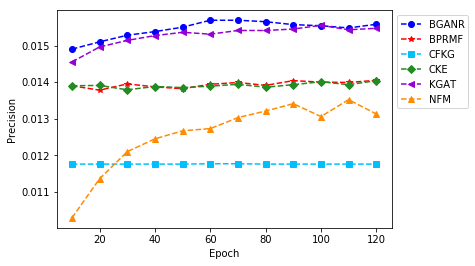
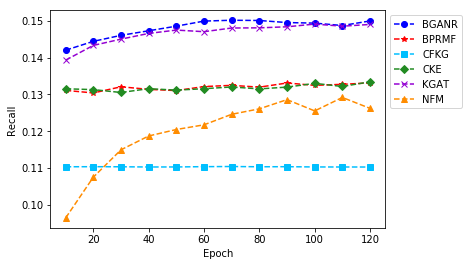


图3 Amazon-book 数据集上不同算法Recall对比 图4 Amazon-book 数据集上不同算法Precision对比

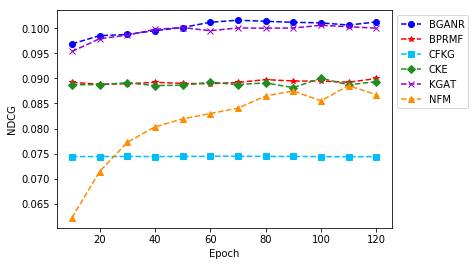
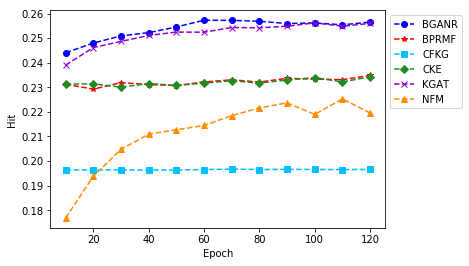


图5 Amazon-book 数据集上不同算法Hit对比 图6 Amazon-book 数据集上不同算法NDCG对比

Last-FM，这里需要对图进行描述，需要一段话。

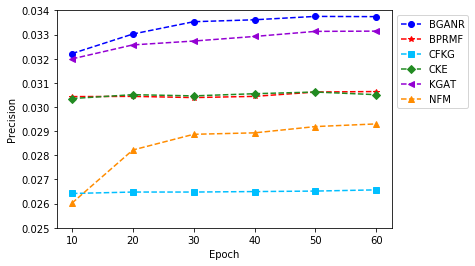
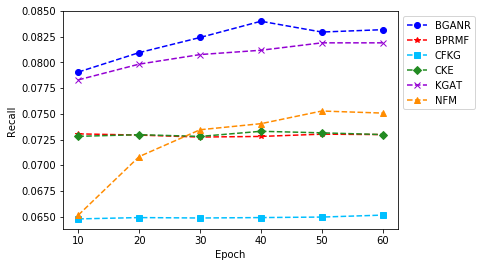


图7 Last-FM 数据集上不同算法Recall对比 图8 Last-FM 数据集上不同算法Precision对比

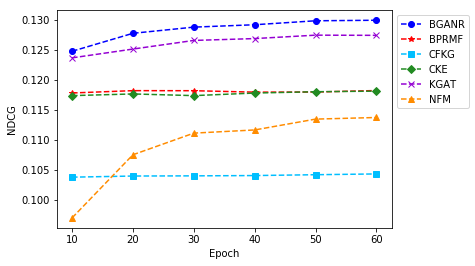
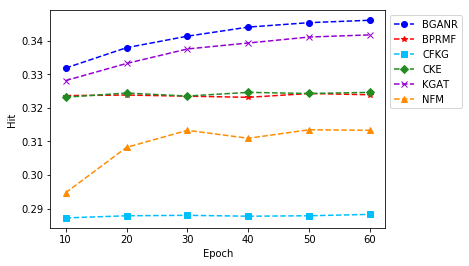


图9 Last-FM 数据集上不同算法Hit对比 图10 Last-FM 数据集上不同算法NDCG对比

BAGNR在两个数据集上产生最佳性能。特别是在Last-FM数据上Recall、Precision、Hit以及NDCG指标分别提高了2.56%、1.87%、1.28%、1.96%。通过叠加双向注意力嵌入传播层，能够以显式的方式双向探索高阶连通性，以便捕捉有效的协同信号。这验证了模型可以捕获协作信号以传递知识。此外，与KGAT相比，BAGNR证明了该方法的有效性，其可以进行双向信息挖掘，而不是单方向。

这里需要对表2进行描述，需要一段话。

表2 算法指标比较

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Amazon-Book | | | | Last-FM | | | |
|  | Recall | Precision | Hit | NDCG | Recall | Precision | Hit | NDCG |
| BPRMF | 0.13314 | 0.01405 | 0.23480 | 0.08997 | 0.07305 | 0.03063 | 0.32432 | 0.11824 |
| CFKG | 0.11043 | 0.01177 | 0.19671 | 0.07449 | 0.06515 | 0.02657 | 0.28825 | 0.10432 |
| CKE | 0.13344 | 0.01404 | 0.23429 | 0.09003 | 0.07330 | 0.03061 | 0.32466 | 0.11814 |
| NFM | 0.12918 | 0.01351 | 0.22530 | 0.08856 | 0.07527 | 0.02930 | 0.31350 | 0.11374 |
| KGAT | **0.14911** | **0.01555** | **0.25622** | **0.10059** | **0.08190** | **0.03313** | **0.34181** | **0.12747** |
| BGANR | **0.15015\*** | **0.01569\*** | **0.25728\*** | **0.10159\*** | **0.08400\*** | **0.03375\*** | **0.34618\*** | **0.12997\*** |
| %Improv | 0.69% | 0.90% | 0.41% | 0.99% | 2.56% | 1.87% | 1.28% | 1.96% |

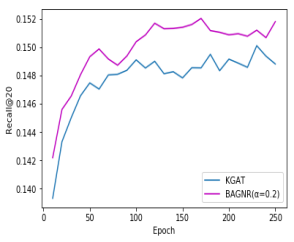
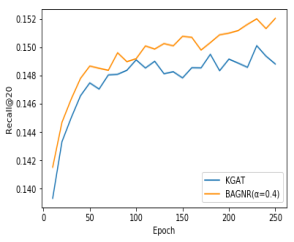
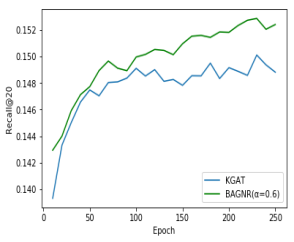
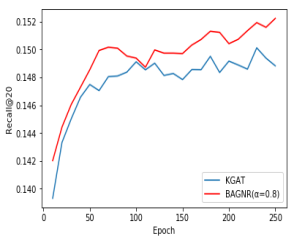
**4.4 消融实验及最优参数确定**

选取适合模型的参数是提高模型表现的一个重要指标，本算法通过在Amazon-Book数据集进行消融实验来确定和**β**的最优值，在Last-FM数据集进行验证。

**4.4.1 最优α**

为了寻找合适的参数值，首先将设置β为0.5，对于*PReLU*激活函数来说，是一个相对对称的一个参数。并在{0.8,0.6,0.4,[0.2](mailto:0.2下的BGANR算法与KGAT算法Recall@20)}调整参数。在Amazon-Book数据集下经过大量的实验，并与KGAT算法进行对比如图(11)-(14)所示。其中各个分图的参数为β=0.5，α分别从{0.8,0.6,0.4,[0.2](mailto:0.2下的BGANR算法与KGAT算法Recall@20)}集合中对应取值。在各个指标的对比图中可以观测到，随着迭代次数的增加，BGANR算法表现优秀，并在α=0.8时，算法表现性能最佳。

图11 Recall@20指标下*α*的最优值



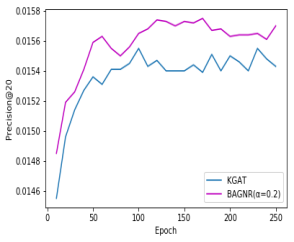
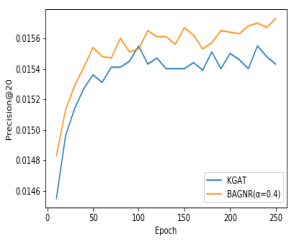
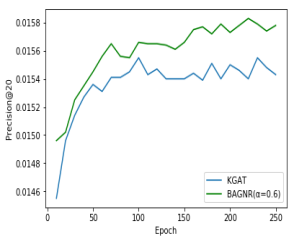
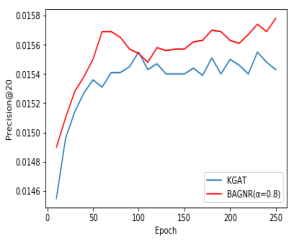


图12 Precision@20指标下*α*的最优值

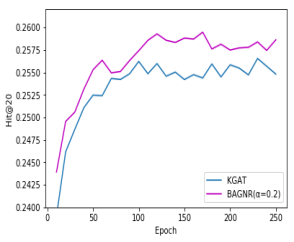
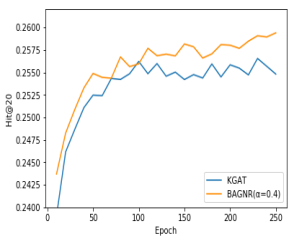
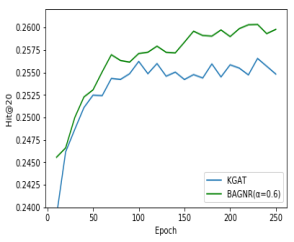
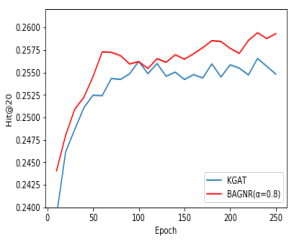


图13 Hit@20指标下*α*的最优值

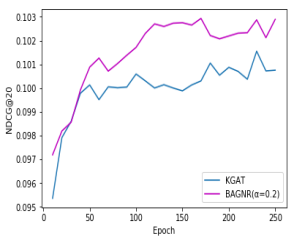
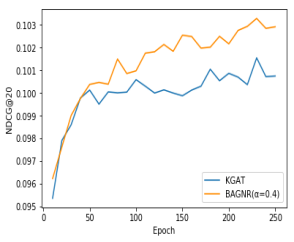
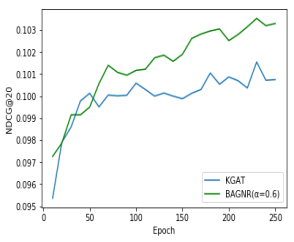
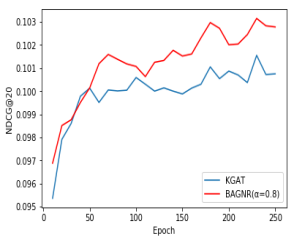


图14 NDCG@20指标下*α*的最优值

表3是在设置不同*α*值下，不同指标的最优结果，其中第二行Origin是原始KGAT算法的各项指标，从表格中可以看出，当*α*=0.8，有四项指标是最优的，最终选择α=0.8为最佳参数，在最下面一行，给出了每项指标相对于原始KGAT算法改善的百分比。

表3 Amazon-book数据集下不同*α*取值指标比较

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Recall@20 | recall@100 | precision@20 | precision@100 | Hit@20 | Hit@100 | NDCG@20 | NDCG@100 |
| Origin | 0.15001 | 0.31689 | 0.01555 | 0.00727 | 0.25655 | 0.48309 | 0.10155 | 0.15561 |
| 0.8 | **0.15315** | 0.31860 | **0.01583** | 0.00730 | **0.26044** | 0.48775 | **0.10362** | 0.15731 |
| 0.6 | 0.15287 | 0.31852 | 0.01579 | 0.00733 | 0.26033 | **0.48822** | 0.10353 | **0.15748** |
| 0.4 | 0.15214 | 0.31819 | 0.01572 | 0.00729 | 0.25866 | 0.48707 | 0.10320 | 0.15710 |
| 0.2 | 0.15204 | **0.31940** | 0.01575 | **0.00735** | 0.25947 | 0.48802 | 0.10293 | 0.15715 |
| Best α = 0.8 | 2% | 0.5% | 1.8% | 0.4% | 1.5% | 0.9% | 2% | 1% |

**4.4.2 最优β**

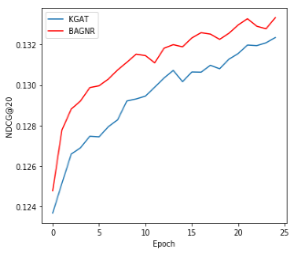
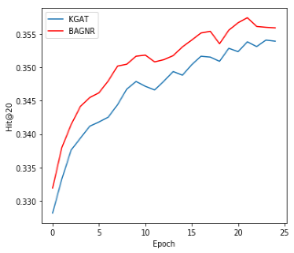
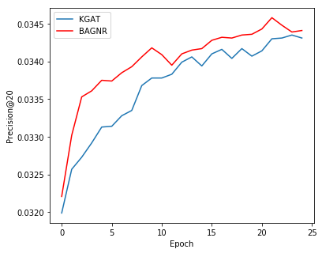
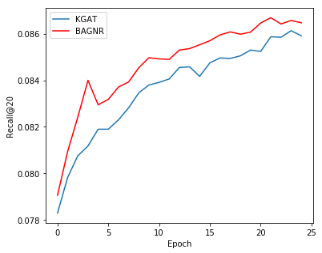
在上述的实验找到了合适的，在本节固定的值，调整β的值以寻找合适参数，选取了三个值进行了实验，分别是0.3、0.5以及0.7。在Amazon-Book数据集进行实验。实验结果如表4所示，通过对比发现，当β= 0.5时各个指标的效果最好。

表4 Amazon-book数据集下不同β取值下指标的对比

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Recall@20 | Recall@100 | Precision@20 | Precison@100 | Hit@20 | Hit@100 | NDCG@20 | NDCG@100 |
| β=0.3 | 0.15052 | 0.31826 | 0.01572 | 0.00730 | 0.25909 | 0.48683 | 0.10296 | 0.15693 |
| β=0.5 | **0.15015** | **0.31858** | **0.01583** | **0.00730** | **0.26044** | **0.48775** | **0.10362** | **0.15731** |
| β=0.7 | 0.14992 | 0.31858 | 0.01575 | 0.00730 | 0.25968 | 0.48836 | 0.10297 | 0.15700 |

**4.4.3验证模型**

在4.4.1和4.4.2中，通过多次实验找到了适合模型的参数和β，为了验证提出的模型的可行性和准确性，我们在Last-FM数据集将本文提出的算法BAGNR与4.3中最优的模型KGAT进行对比，进而可以验证了BAGNR的可行性。各项指标对比结果如图(19)所示，BGANR算法优于KGAT算法，这也就验证了BGANR算法选择参数的有效性和算法可行性。



(a)Recall@20 (b)Precision@20 (c)Hit@20 (d)NDCG@20

图19不同指标下KGAT与BAGNR的对比

表5是在Last-FM数据集下KGAT与BAGNR算法的各项指标的对比，从图中可以看出，BGANR算法明显优于KGAT算法

表5 Last-FM数据集下KGAT与BAGNR各项指标的对比

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Recall@20 | Recall@100 | Precision@20 | Precison@100 | Hit@20 | Hit@100 | NDCG@20 | NDCG@100 |
| KGAT | 0.08794 | 0.17009 | 0.03468 | 0.01727 | 0.36048 | 0.60333 | 0.13526 | 0.22021 |
| BAGNR | **0.08820** | **0.17082** | **0.03521** | **0.01741** | **0.36111** | **0.60549** | **0.13660** | **0.22085** |
| %Improv | 0.81% | 0.43% | 1.53% | 0.81% | 0.17% | 0.35% | 0.99% | 0.29% |

综合对比图与表格可以发现，在实际生活中的数据集中，我们提出的改进的方案是可行。

**5 结论与展望**

本文在基于知识图谱的推荐算法基础上，在知识嵌入层，提出一种双向知识表示的嵌入方法，不仅考虑到从头实体到尾实体的距离，而且还考虑到从尾实体到头实体的距离。在传播层，首先获取项目与用户间的高阶关系，然后通过注意力机制来定义不同路径的权重关系。在信息聚合层，根据高阶关系设置PReLU激活函数不同通道避免梯度消失和过拟合。在Amazon-Book和Last-FM两个公共的数据集验证了本文提出算法的有效性。在未来的工作中，一方面将考虑异构数据间的用户/项目的高阶路径的获取，另一方面在构造损失函数中，将考虑使用自适应性Dynamic ReLU作为激活函数。并将时序作为精度推荐的依据，进一步挖掘在不同时间段推荐的结果，提高推荐精度。

**参考文献[需要从新改参考文献]**

[1]Travis Ebesu, Bin Shen, and Yi Fang. 2018. Collaborative Memory Network for Recommendation Systems. InSIGIR. 515–524

[2]Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-SengChua. 2017. Neural Collaborative Filtering. InWWW. 173–182

[3]Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Latent relational metric learningvia memory-based attention for collaborative ranking. InWWW. 729–739

【W1-2020-10-5 17:07:09】[4]Y. Koren. Factorization meets the neighborhood: Amultifaceted collaborative filtering model. InKDD, pages426–434, 2008

[5]H. Wang, N. Wang, and D.-Y. Yeung. Collaborative deeplearning for recommender systems. InKDD, pages1235–1244, 2015

[6]S. Rendle. Factorization machines. InICDM, pages995–1000, 2010.  
[7]Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-SengChua. 2017. Neural Collaborative Filtering. InWWW. 173–182

[8]JingyuanChen,HanwangZhang,XiangnanHe,LiqiangNie,WeiLiu,andTatSengChua.2017. AttentiveCollaborativeFiltering:MultimediaRecommendation withItem-andComponent-LevelAttention.InSIGIR.335–344.

[9]Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All youNeed.InNeurIPS.6000–6010.

[10]Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification

[11] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep Neural Networks forYouTube Recommendations. InRecSys. 191–198.

[12] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L. Hamilton,and Jure Leskovec. 2018. Graph Convolutional Neural Networks for Web-ScaleRecommender Systems. InKDD (Data Science track). 974–983

[13]Yehuda Koren, Robert M. Bell, and Chris Volinsky. 2009. Matrix FactorizationTechniques for Recommender Systems.IEEE Computer42, 8 (2009), 30–37.

[14] Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Latent relational metric learningvia memory-based attention for collaborative ranking. InWWW. 729–739.

[15] Marco Gori and Augusto Pucci. 2007. ItemRank: A Random-Walk Based Scoring Algorithm for Recommender Engines. InIJCAI. 2766–2771.

[16]William L. Hamilton, Zhitao Ying, and Jure Leskovec. 2017.InductiveRepresentation Learning on Large Graphs. InNeurIPS. 1025–1035.

[17]Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification withGraph Convolutional Networks. InICLR.

[18] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. 2014. SpectralNetworks and Locally Connected Networks on Graphs. InICLR.

[19]Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. 2016.Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering.InNeurIPS. 3837–3845.

[20]Thomas N. Kipf and Max Welling. 2017. Semi-Supervised Classification withGraph Convolutional Networks. InICLR.

[21—N\_2020/10/5]WayneXinZhao,GaoleHe,Hong-JianDou,JinHuang,SiqiOuyang,andJi-Rong

Wen.2018. KB4Rec:ADatasetforLinkingKnowledgeBaseswithRecommender Systems. CoRR abs/1807.11141(2018).