**用于二部图嵌入的异构图卷积网络**

**摘要**

二部图嵌入的研究相对较少。可以证明，一般的图嵌入方法如node2vec和LINE也可以忽略顶点特征来学习二部图嵌入。然而，现实世界的二部图也关注顶点特征以及两种实体之间的显式和隐式关系，这两种实体通常表现出不同于其他图形结构数据类型的特性和模式。同时，由于二部图的异构性，基于卷积的模型如GCN和R-GCN不能适用于二部图。为了解决这些问题，本文首先提出了异构图卷积网络(HGCN)来学习二部图的图表示。在此基础上，我们建立了三级级联的HGCN体系结构和两个输出模型，包括解码器模型和对抗学习模型。该体系结构可以学习二部图中两个组之间的显式关系和隐式关系。通过对两个大型二部图数据集的评价表明，HGCN比Node2Vec和DeepWalk等方法具有更好的学习表示性能。

**1介绍**

图提供了一个无处不在的数据结构去模拟实体(即顶点)之间的交互（例如,边缘),被广泛应用于许多应用程序,如社交网络(王et al ., 2017 b),知识图(曹et al ., 2017),推荐系统(他et al ., 2015),和其他(冯et al ., 2018)。到目前为止，大多数研究工作都集中在图形嵌入上。

特别地，它们将一个顶点表示为一个可学习的嵌入向量，其中向量之间的接近编码了关于图拓扑的信息。因此，顶点嵌入可以被输入到机器学习方法中，以解决各种任务，如分类、排序、链接预测、聚类、可视化等。

然而，他们只处理像社交网络一样顶点类型相同的同质网络，(Perozzi等人，2014;Zhu等人，2018;Liao等人，2018;Li等人，2017)，或像知识图这样顶点(和/或边)是不同类型的异构网络(Chang等人，2015;Dong等人，2017;Xu等，2017)。对二部图的研究相对较少。

虽然一般的图嵌入方法如DeepWalk (Perozzi et al.， 2014)和node2vec (Grover和Leskovec, 2016)可以通过忽略顶点类型信息来学习二部图的顶点嵌入。但是这些方法在这样做时并不理想，原因有几个:

(1)由于现实世界的二分网络涉及两种类型实体之间的关系，显式关系(即观察到的链接)和隐式关系(即未观察到但可传递的链接)，它们通常表现出不同于其他类型网络数据的属性和模式。例如，社交网络系统需要捕捉用户之间的关系以及他们与聊天组或爱好社区的关系，电子商务推荐系统需要捕捉客户和产品之间的协同过滤模式，搜索引擎需要考虑查询和网页之间的匹配信号。

(2)幂律分布是现实世界中很多二部图网络的共同特征(Pongnumkul和Motohashi, 2018)，但由通用随机游走算法生成的语料库可能无法保持这一特性，如DeepWalk中使用的。这可能会限制高度数顶点的信息，并对低度数顶点进行过采样。

我们的工作填补了异构图卷积网络二部图学习图表示的研究空白。HGCN是我们为二部图嵌入提出的一个新模型，它解决了以下挑战:

1. 大规模的图。节点/边数大于100万。
2. 异构边缘/节点功能。边缘和节点包含不同的特征向量。主要的难点是如何将来自不同空间的不同信息表示为一个统一的嵌入空间。
3. 半监督标签，甚至无标签的无监督学习。大规模图中节点的手工标注需要大量的人力资源，在大多数情况下甚至是不可能完成的。
4. 显性和隐性关系。现实世界的二部图关注两类顶点之间的关系。我们的

GCN模型应该用一个联合优化框架同时建模显式和隐式关系。

**2相关的工作**

我们的工作与神经图嵌入方法有关。我们首先从网络类型的角度回顾它们。

DeepWalk (Perozzi et al.， 2014)和Node2vec (Grover and Leskovec, 2016)将Skip-gram (Mikolov et al.， 2013)的思想扩展到同质网络模型。然而，它们可能不能有效地保持图的显式和隐式关系。后续还有一些工作是利用顶点之间的一阶和二阶近似来嵌入同质网络。具体来说，LINE (Tang et al.， 2015)学习了一阶和二阶关系的两个分离嵌入;SDNE (Wang et al.， 2016)和DVNE (Zhu et al.， 2018)合并了一阶和二阶近似以保持网络结构;GraRep (Cao et al.， 2015)和AROPE (Zhang et al.， 2018)进一步扩展了该方法以捕获高阶近似。除了获取高阶近似外，还有几种将边信息并入顶点嵌入学习的建议，比如顶点标签(Li等人，2017;Huang et al.， 2017)、社区信息(Chen et al.， 2016)、文本内容(Wang et al.， 2017a)、用户概况(Liao et al.， 2018)和位置信息(Xie et al.， 2016)。然而，由于忽略了顶点类型信息，这些方法对于学习二部图网络的顶点表示可能是次优的

Metapath2vec++ (Dong et al.， 2017)、HNE (Chang et al.， 2015)和EOE (Xu et al.， 2017)是异构网络中代表性的顶点嵌入方法。虽然二部图可以看作是一种特殊类型的异构网络，它们可以应用于二部图，但它们并不是为在二部图网络上学习而量身定制的。具体来说，HNE旨在将内容和链接结构集成到嵌入过程中，而metapath2vec++忽略了顶点之间关系的强度，将显式关系和隐式关系同等对待。因此，它们对于二部图网络的顶点表示学习是次优的。值得注意的是，最近的研究显示，人们对通过邻域聚合编码器生成顶点嵌入越来越感兴趣(Kipf和Welling, 2016;Schlichtkrull等人，2017a)。这些方法依赖于顶点特征或属性。然而，他们没有考虑长尾/幂律分布问题。

二部图网络作为一种普遍存在的数据结构，已被挖掘到许多应用领域，其中顶点排序是一个活跃的研究问题。例如，HITS (Kleinberg, 1999)通过捕获二部图网络中的一些语义关系来学习对顶点进行排序。Co-HITS (Deng et al.， 2009)将顶点的内容信息和关联约束引入二部图的顶点排序中。BiRank (He et al.， 2017)通过同时考虑网络结构和先验知识对顶点进行排序。分布顶点表示是利用二部图信号的另一种方法。与排序任务不同的是，它学习一个顶点的低维表示，这可以被看作是保存更多信息的顶点的“特征”，而不仅仅是一个排序分数。

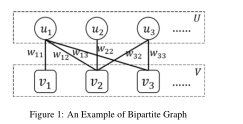
潜在因素模型(LFM)是推荐系统和语义分析领域中最具代表性的模型，在推荐系统和语义分析领域得到了广泛的研究。LFM的一个典型实现是基于矩阵分解(Rendle等人，2009;Zhang等人，2016;He等人，2016)。最近的进展利用深度学习方法学习用户物品网络的顶点嵌入进行推荐(He和Chua, 2017)。值得指出的是，这些方法是为推荐任务量身定制的，而不是为学习信息性顶点嵌入而定制的。此外，他们只对二部图网络中的显式关系建模，可以通过合并隐式关系加以改进，如(Jiang et al.， 2016;Y u等，2018a)。

由于许多有趣的任务涉及到的数据不能简单地用欧氏空间表示，而是存在于不规则的图形状结构中，如社会网络、生物网络或大脑连接。因此，为了扩展神经网络来处理任意结构的图，早期的工作使用递归神经网络来处理图域中表示为有向无环图的数据(Frasconi et al.， 1998;Sperduti和Starita, 1997)。图神经网络(GNNs)首次被引入(Gori等人，2005;Scarselli等人，2009)作为递归神经网络的一种推广，它可以直接处理更一般的一类图。在卷积网络在计算机视觉领域取得巨大成功的驱动下，许多方法对图数据应用了相同的卷积表法，其中称为图卷积网络(GCN) (Wu et al.， 2019;Kipf和Welling, 2017)。这方面的进展通常分为光谱方法和非光谱方法。谱方法与图的谱表示一起工作，其中学习滤波器依赖于拉普拉斯特征基(Bruna等人，2013;Henaff等人，2015)。对于非谱方法，(Hamilton et al.， 2017;Duvenaud等人)卷积直接定义在图上，操作在空间上相邻的组。一般来说，这些方法通过聚合相邻节点的信息来执行卷积，因此每个节点都能够了解整个图中的节点之间的关系。基于GCN的这一特性，图自动编码器(GAE)被引入来解决网络嵌入问题，其目的是将网络顶点表示为一个低维向量空间(Cao等;Wang等人，2016;Pan等人，2018;Y u等，2018b)。其思想是，编码器首先利用图形卷积层将图形嵌入到一个潜在空间中，在该潜在空间上，解码器用于重建图形结构。然而，基于GCN的方法的缺点是应该假定相同的节点属性。除了基于GCN的框架之外，另一种解决方案是直接使用堆叠式自动编码器提取每个节点的隐藏表示。然而，同样的设置也存在，不同的节点应该具有相同的属性，这在二部图中是不可用的。(Nassar, 2018)试图将GCN与二部图结合起来，他们通过聚类聚合节点，生成可以有效加速和扩展GCN算法计算的二部图，但他们的目标并不是学习二部图数据的表示。

**3异构图二部图的卷积网络**

二部图具有不同的特征空间，但两组之间有很强的联系，这促使我们使用图卷积网络GCNs (graph convolution networks)来学习二部图嵌入。与纯粹基于内容的深度模型不同，GCNs和R-GCN (Schlichtkrull等人，2017b)既利用了内容信息(如顶点的特征)也利用了图的拓扑结构。而在二部图网络中，互不连接的两个顶点集往往具有不同的表示和特征空间，例如在推荐系统中，一个顶点集是用户表示，另一个顶点集是项特征空间。这意味着，在大多数情况下，二部图有异构的顶点，其中直接聚合邻居的特征与自身是不可能的。现有基于卷积的方法(如GCN和R-GCN)的缺点促使我们设计新的GCN模型。主要难点是如何将来自不同空间的不同信息(来自异构顶点的特征)表示为一个统一的嵌入空间。

**3.1问题定义**



定义1。设G = (U, V, E)是二部图网络，其中U和V分别表示两种类型的顶点集合，E⊆U×V定义集间边。Ui和Vj分别表示U和V中的第i个和第j个顶点，其中i = 1,2，…， M和j = 1,2，…，N。因此节点集U的邻接矩阵为A，A是一个满足“如果eij∈E，则Aij= wij > 0；如果eijE，则Aij= 0”的M\*N维的矩阵。节点集B的邻接矩阵就是AT∈RN×M。

二部图中的两个节点集可以分别与节点属性xu和Xv相关联，其中，Xu∈RM×P为特征矩阵，xui RP表示节点ui的特征向量；同理，Xv∈RN×Q也是一个特征矩阵，Xvj∈RQ表示节点vj的特征向量。

异质二部图数据中表示学习的任务旨在将图中的所有节点映射到低维嵌入空间，其中每个节点都表示为密集的嵌入向量。在嵌入空间中，此嵌入向量应在无标签的无监督学习设置中保留以下信息：

1. 两个不同类型节点的特征以及两个不同类型节点之间的边缘权重/特征。
2. 相同类型的节点之间的隐式关系。
3. 不同类型节点之间的显式关系。

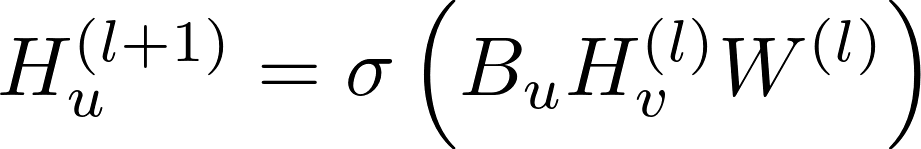
**3.2异构图卷积网络（HGCN）**

传统上，对同构图执行的卷积定义为



其中，A是邻接矩阵，D是对角节点度矩阵，H（l）和W（l）分别是第l层隐藏表示和权重矩阵。 σ（·）表示激活函数，例如ReLU（·）= max（0，·）。对于异质二部图，由于两个节点集的特征表示形式不同，因此无法直接使用其自身聚合附近的特征。

为了解决这个问题，我们提出了一种新颖的方法，该方法首先通过卷积运算合并附近节点的特征，然后通过输出模型将其自身合并。使用输出模型来组合节点属性和结构嵌入的原因是，节点的属性与其与不相交集合中的节点的连接及其特征高度相关，例如，在我们的数据集中，黑人用户可能共享相同的特征，并且与某些组具有相同的连接。使用输出模型将会以非线性方式结合属性和拓扑结构。具体来说，用于计算集合U中节点的前向传递更新的传播模型是:



其中。是集合U的邻接矩阵。是集合U的用于归一化的对角度数矩阵。是集合U的隐藏层表示。

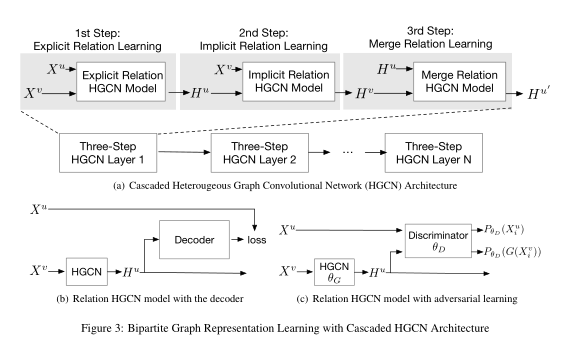
当l = 0时，则，这是集合V的输入特征。同样，对于集合V中的节点，更新公式为:



其中是集合V的对角度数矩阵。是集合U的隐藏层表示。

不同于常规的遗传神经网络，我们引入了异构变换，即首先根据类型或特征空间聚集附近节点的特征向量，然后通过输出模型对齐其自身的特征。这有助于学习一种既包含节点特征又包含邻居节点特征和网络结构信息的表示法。我们提出了两个具体的输出模型，这将在接下来的章节中介绍。

我们将上述传播模型和输出模型称为异构图卷积网络(HGCN)，如图所示。



**3.3级联HGCN架构**

我们设计了一个级联的HGCN架构(图3(a))，这是一个无监督的学习模型。其动机是大规模图形中节点的手工标注需要相当多的人力资源，在大多数情况下甚至是不可能的。除此之外，我们旨在学习包含节点特征信息以及隐含和显式网络信息的节点表示。

如图3(a)所示，它包含三个步骤:显式关系学习步骤、隐式关系学习步骤和合并学习步骤。级联HGCN架构的第一步是明确的学习步骤。其目的是学习组U的节点表示，以便它包括组U的特征信息，还包括组V的邻居特征和网络拓扑信息。这个第一步的输入是组U的特征Xu，和对面组V的特征Xv，输出是隐藏表示Hu。

我们很容易发现隐藏表示Hu不包含间接的网络拓扑关系，即U组中的节点通过V组间接连接到同一组中的其他节点，因此我们设计了第二步:隐含关系学习步骤。它采用第一步的隐藏表示Hu和V组的Xv作为输入。类似于步骤1的输出，输出表示Hv包含组V的特征信息，并且还包括其组U邻居特征和网络拓扑信息。第三步使用显式表示Hu和隐式表示Hv输入，因此输出将包含隐式和隐式表示。

我们可以把三步级联的HGCN架构看作一个网络层。如图3(a)所示，我们可以基于三步架构设计一个多层网络架构。级联HGCN架构的训练过程是一步一步，然后一层一层。它的最终输出是Hu’，Hu’既可以包含当前组和其相对组的特征，也可以包含显式和隐式网络信息。

HGCN可以看作是编码器-解码器模型，其中编码器既编码图形拓扑，又编码不相交集中的特征信息。解码器将这种隐藏的表示与节点自身的特征进行映射，因为我们假设一个节点的连接与其自身的特征高度相关。为了对齐节点特征Xu和隐藏Hu之间的相似性，我们设计了两个输出模型。第一个是带有解码器模型的HGCN，第二个是带有对抗性学习模型的HGCN。在图3（a）中，每个步骤的HGCN模型都可以选择带有解码器模型的HGCN或带有对抗学习模型的HGCN。在我们的实验部分，我们将比较这两种输出模型的性能。

**3.4带解码器模型的HGCN**

带解码器模型的HGCN如图3(b)所示。首先基于(2)计算一个前向通路，得到V的拓扑和节点表示，即Hu。然后将解码器应用于，试图将与其自身属性Xu对齐。这里我们使用一个简单的单层神经网络作为解码器:



因此，集合U的损失函数可以定义为:



使用神经网络作为解码器而不是简单地连接两个表示向量的原因是，我们希望以非线性方式将拓扑表示与节点自身的特征对齐，以便最终的表示可以嵌入这两个边信息。因此，集合V的解码器和损失函数是相同的：

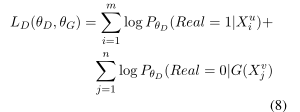


如果第二步和第三步选择该解码器模型作为输出模型，则应用相同的公式和损失函数，但是每个输出模型具有不同的参数。

**3.5带有对抗性学习的HGCN**

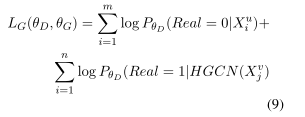
我们为每一步设计的另一个输出模型是带有对抗性学习的HGCN模型，其中我们将学习过程定义为一个两人游戏。以图3(c)为例，它的第一个游戏玩家是HGCN块，其目的是最大化U组的特征Xu与其输出Hu之间的相似度，Hu是取V组的特征Xv作为输入时HGCN操作的结果。另一个玩家是鉴别者，他的目的是区分它是来自真正的Xu特征还是通过HGCN学习的Hu特征。鉴别器的输入来自真实的节点特征Xu和混合高斯核函数的输出Hu。这两个玩家形成了一个敌对的环境，每个人都最大限度地发挥了自己的能力。这种方法符合Ganin等人(2016)的工作，他们提出学习对输入域不变的潜在表示，在我们的情况下，域是二分图中两个强相关的组。

鉴别器目标。我们将鉴别器的参数定义为θD。它的损失函数如下:



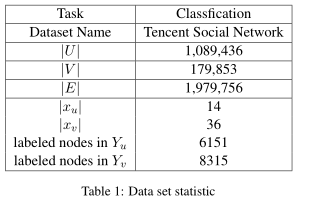
其中“ real = 1”表示鉴别器的输入样本来自其自身特征的表示，而“ real = 0”表示鉴别器的输入样本是由HGCN生成的隐藏表示

HGCN目标。其损失函数如下：



为了训练对抗学习模型，我们遵循Goodfellow等人的深度对抗网络的标准训练程序。 （2014）。对于每个输入样本，使用随机梯度更新对鉴别器和HGCN进行连续训练，以分别最小化LD和LG。

**4实验**



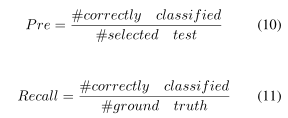
为了评估表示性能，我们首先对U组进行表示学习，然后基于U中每个节点的表示结果，使用一个简单的分类器（我们使用在实验中使用了逻辑回归）用于评估节点表示的性能。对于基于随机游走的模型，我们选择DeepWalk和Node2V ec作为与HGCN模型的比较。对于基于卷积的模型，我们比较了本文提出的三种HGCN模型。

•一层HGCN（1L-HGCN）。该模型首先在组U中的节点上执行HGCN的单层以获得隐藏向量Hu，然后将特征Hu与Xu连接到输出节点表示。

•带解码器的三层HGCN（3LDec-HGCN）。如图（a）所示，该模型是三层级联HGCN模型，具有图（b）所示的Decoder模型。

•具有对抗性学习的三层HGCN（3L-Adv-HGCN）。如图（a）所示，此模型是三层级联HGCN模型，其对抗性学习模型如图（c）所示。

我们的问题是分类问题，在我们的数据集中是对一个人或一个团体是否来自黑市进行分类。为了测量和评估不同方法的性能，在这里我们选择精度（Pre）和召回率作为我们的评估指标。公式为：



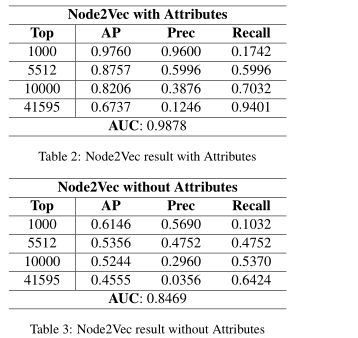
给定一个样本的精度表征，正确分类的概率是多少。召回确定了可以找到模型的地面真理的百分比。同样，平均精度（AP）用于总结在每个阈值处获得的精度的加权平均值，而前一个阈值的召回率增加用作权重：



其中Pn和Rn是第n个阈值的精度和召回率。

4.2实验结果表

表2总结了性能比较。我们已经完成了Node2V ec和原始特征模型的实验。



Node2V ec (Grover and Leskovec，2016)是一种著名的节点嵌入方法，在图中执行偏置随机行走来捕获拓扑结构信息。基线实验首先使用Node2Vec得到每个节点的节点嵌入。然后使用逻辑回归来确定用户或组是否来自黑市。在这里，我们实现了逻辑回归，添加和不添加节点自己的属性。从表2和表3中我们可以清楚地看到，通过添加节点的属性，模型可以在分类中表现得更好。因此，不仅是图的拓扑结构，节点的特征在分类中也很重要。这进一步激励我们结合多种类型的特征向量和图形拓扑。此外，根据top-k排名，精度和召回率呈负相关，精度越高，召回率越低。

注:截至中期截止日期，我们已经完成了DeepWalk和Node2V ec的基线评估。1L-HGCN模型正在开发中。我们差不多完成了，但还没有结果。其他实验预计在最终项目报告中完成。本文的代码维护在GitHub上。GitHub地址是:https://github.com/chaoyanghe/bipartitegraph-learning.我们已经添加了TA (GitHub ID:Hunter-Lin)作为我们的合作者。请检查代码以检查我们的进度。

**5结论**

正在进行中。

**鸣谢**

正在进行中。