山东大学

毕业论文(设计)

论文题目：符号网络的社团划分研究

姓　　名 王云平

学　　号 201900820077

学　　院 数学与统计学院

专　　业 统计学

年　　级 2019

指导教师 亓兴勤

2023 年 5 月 2 日

摘　要

符号网络指的是个体之间拥有正、负两种关系的网络，作为复杂网络的分支之一，是复杂网络的一种推广和特殊形式。对于一般的复杂网络来说，社团内部比较紧密，社团之间相对比较松散，而对于符号网络来讲，社团内部拥有尽可能多的正边，社团之间拥有尽可能多的负边。确定和了解符号网络中的社团结构，会更有利于我们认识和理解真实的社会网络，从而能够更加深入地分析研究社会网络中的信息传递模式等等，增加行为预测等实际应用的准确性。

图神经网络是一种人工神经网络，与传统的神经网络模型不同，图神经网络可以利用节点之间的关系信息来处理图结构数据，提取图结构的特征，从而实现更准确的预测和分类。图神经网络已经在许多领域取得了显著的成功，如社交网络分析、化学物质分析和基因调控预测等。

本文的主要工作有：综述了几种业界已有的符号网络社团划分算法的基本流程原理；提出了一种基于符号网络上的图卷积神经网络模型进行节点聚类以实现社团划分的方法；同时，我们还提出了一种基于符号网络上的节点相似性进行谱聚类的社团划分算法。

关键词: 符号网络；社团划分；社区检测；图神经网络；图卷积神经网络

ABSTRACT

Signed networks refer to networks with positive and negative links between individuals. As a branch of complex networks, they are a generalization and special form of complex networks. For general complex networks, the nodes links within the communities are relatively tight and the nodes links between them are relatively loose, while for signed networks, nodes in communities have as many positive edges as possible and nodes between communities as many negative edges as possible. Identifying and understanding the community structure in signed networks will be more conducive to our understanding and understanding of real social networks, enabling us to conduct more in-depth analysis and research on information transmission patterns in social networks, and increase the accuracy of practical applications such as behavior prediction.

Graph neural network is an artificial neural network that, unlike traditional neural network models, can use the relationship information between nodes to process graph structure data, extract features of graph structure, and achieve more accurate prediction and classification. Graph neural network has achieved remarkable success in many fields, such as social network analysis, chemical analysis and gene regulation prediction.

The main work of this article includes: summarizing the basic process principles of several existing signed network community partitioning algorithms in the industry; and proposing a node clustering method based on graph convolutional neural network models on signed networks to achieve community partitioning; At the same time, we also propose a community partition algorithm based on node similarity in signed network for spectral clustering.

Key words: signed network, community detection, graph neural network, graph convolutional network

目 录

Contents

第一章 引言

1.1 研究背景与研究意义

随着科技的不断发展与文明水平的不断提高，表征错综复杂社会关系的复杂网络研究引起了众多研究者的关注，与此相关的网络相关研究也取得了极大的进步，深入开展社会网络结构的研究的作用也显得越来越重要。社团研究作为网络研究的一个重要的组成部分，对于了解网络结构与增强其稳定性等方面具有重要的意义。社团划分指的是将网络划分为几个社团，使得社团内部的节点之间联系紧密，而社团之间的节点联系相对稀疏。

符号网络是复杂网络中相当重要的一个分支，表示网络之间的连接有正边与负边的区别，代表个体的网络节点之间不仅有着正向关系，也有着负向关系。比如，在社交网络中，两人之间既可能是朋友关系，也可能是敌对关系，而这两种关系所表示的社会意义不同，显然是不太能够一概而论的。于是对于符号网络而言，我们进行社团划分时就不能仅考虑节点间联系是否紧密的问题，而应当同时考虑节点间连边的符号：也即使得在社团内部的节点之间的连边符号尽可能为正，社团之间的节点连边符号尽可能为负。

FEC 算法[1]是符号网络社团划分的一个比较经典的算法。它首先选择网络中的任意一个节点，开始进行随机游走。它会根据邻接矩阵计算出的转移概率分布来选择下一个节点，这个转移概率分布体现的是当前节点与其他节点的连接情况。其理论依据是：根据网络社团的拓扑结构，我们从任意一个节点开始随机游走，并经过多次游走后，停留在同一个社团中的概率大于停留在其他社团中的概率。这也表明，如果两个节点的期望转移概率分布相似，它们可能属于同一个社团。FEC 算法不需要事先了解社团结构, 同时考虑到网络的连边与节点密度, 对于噪声不敏感, 是一种非常高效的算法.

ILPAG[2]是一种基于标签传播算法进行改进的社团划分算法。普通标签传播算法的流程如下：初始化节点标签，对于每个节点，计算它的邻居节点的标签，统计每个标签在邻居节点中出现的次数，将当前节点的标签设置为出现次数最多的标签。如果有多个标签出现次数相同，则可以随机选择一个标签。ILPAG 算法在将该算法应用在符号网络的基础上，基于结构相似度计算符号网络中每个节点的影响力，通过节点排序等方式降低了算法的随机性，并通过节点与超节点结合等手段，提高了算法的稳定性。

对于一般图网络的节点聚类算法的研究是比较充分的，许多研究者将一般网络中的节点相似度，聚类系数，模块度等指标，根据符号网络的特性进行修改，从而将原

聚类算法更改成可适应符号网络的节点聚类算法，从而可以在符号网络上通过节点聚类算法进行社团划分。

以基于节点相似度的密度峰值聚类算法[3]为例，其大致流程为：计算节点间的相似度，生成相似度矩阵，根据相似度矩阵计算节点密度与距离矩阵；将网络中的节点根据节点密度与距离矩阵，选取聚类中心，得到剩余节点集合；对剩余节点集合依照节点密度从大到小进行遍历，按照节点与聚类中心的距离进行归类得到聚类结果。

图卷积神经网络作为一种处理图网络的深度学习模型，可以用于节点分类、图分类和链接预测等任务。由于其强大的表达能力和广泛的应用前景，作为图神经网络的一个分支，正在成为深度学习领域方兴未艾的研究方向之一。

1.2 论文组织结构

本文的主要工作有两个方面。首先，我们提出了两种符号网络上的社团划分算法：其一是基于符号网络的图卷积神经网络进行节点聚类，从而实现网络中的社团划分的算法，其二是基于符号网络中的节点相似度构建新图进行谱聚类，从而实现网络中的社团划分的算法；另外，我们对上述两种社团划分算法完成了代码实现与一些数据集上的效果评估。本文各章节具体内容如下：

- 第一章：引言部分。此章节主要分两部分，首先，我们将介绍符号网络的社团划分问题的研究背景与研究意义, 介绍几种符号网络社团划分的算法; 其次，我们将介绍论文的主要研究内容与组织结构。

- 第二章：理论基础。我们将构建符号网络的社团划分问题的理论基础，包括符号网络的基础概念与特征量，结构平衡理论，以及社团划分问题及其评价标准等等。

- 第三章：图卷积神经网络。我们介绍一些图卷积神经网络相关的基本内容，包括其基本流程，节点的嵌入表示，信息的聚合过程，损失函数等等。

- 第四章：我们所提出的两种算法的介绍。其一是基于符号图卷积神经网络的节点聚类实现社团划分的算法。主要包括：进行邻居信息聚合，我们改进的损失函数，以及对输出的嵌入表示的聚类操作，最后总归为算法的总流程；其二是基于节点相似度的谱聚类算法的介绍。

- 第五章：实验。我们对所提出的两种算法进行编程实现以及结果评估，包括数据集的简单介绍与算法结果的对比。

- 第六章：总结与展望。我们对我们的研究成果进行分析总结，对下一步的工作进行展望。

第二章 理论基础

2.1 符号网络

2.1.1 定义

符号网络指的是节点间的连边存在正负之分的网络，当两个节点的连边关系为积极的“朋友”关系，我们称这两个节点的连边符号为正。反之，两个节点的连边关系为消极的“敌人”关系，我们称连边符号为负。本文主要研究无向的符号网络。我们一般定义符号网络为 $G=(V,E,s)$，其中 $V=\{v_1,v_2,...,v_n\}$ 表示网络中的节点集合，$E=\{e_{ij}\}=\{(v_i,v_j)\}$ 表示节点连边的集合，$s=(+1,-1)$ 表示为连边符号。符号网络的邻接矩阵定义为：

$$a_{ij}=\begin{cases}1, & \text{节点 } v_i \text{ 和 } v_j \text{ 之间的连边为正}\\ 0, & \text{节点 } v_i \text{ 和 } v_j \text{ 之间没有连边}\\ -1, & \text{节点 } v_i \text{ 和 } v_j \text{ 之间的连边为负}\end{cases} \tag{2.1}$$

我们知道对于一般的图网络，节点 v_i 的度为：

$$k_i=\sum_{j=1}^{n}a_{ij} \tag{2.2}$$

对于符号网络，节点 v_i 的度为

$$k_i=\sum_{j=1}^{n}|a_{ij}| \tag{2.3}$$

考虑到边的符号，对节点 v_i，我们有正度与负度：

$$k_i^+=\sum_{j=1}^{n}a_{ij}^+ \quad k_i^-=\sum_{j=1}^{n}a_{ij}^- \tag{2.4}$$

其中，a_{ij}^+ 指的是节点 v_i 与节点 v_j 之间的正边，a_{ij}^- 指的是节点 v_i 与节点 v_j 之间的负边。

2.1.2 结构平衡理论

总的来说，符号网络的平衡理论[4]意味着：

- 我朋友的朋友是我的朋友

- 我朋友的敌人是我的敌人
- 我敌人的朋友是我的敌人
- 我敌人的敌人是我的朋友

根据上述结构平衡理论，我们可以看出，符号网络中，平衡结构有偶数个负边，不平衡的结构有奇数个负边。Tyler Derr 等人在其论文《Signed Graph Convolutional Network》[5] 中提出了符号网络上的平衡路径与平衡集的概念。定义符号网络中有偶数个负边的路径为平衡路径，有奇数个负边的路径为非平衡路径。

假设从节点 v_i 出发的平衡路径到达的终点 v_j 与 v_i 有边相连，如果这个结构为平衡结构，我们便能得出点 v_i 与 v_j 之间以正边相连。于是，我们可以推知 v_i 与 v_j 之间的关系很可能是积极的。类似地，我们可以推知从节点 v_i 出发的非平衡路径到达的终点 v_j 与 v_i 之间的关系很可能是消极的。

据此，定义从节点 v_i 处，以长度 l 的平衡路径可以到达的节点集合为 $B_i(l)$，我们称为节点 v_i 的 l 阶平衡集；类似地，在节点 v_i 处以长度 l 的非平衡路径可以到达的节点集合为 $U_i(l)$，我们称为节点 v_i 的 l 阶非平衡集。于是我们可以得到：

$$B_i(1) = \{v_j | v_j \in N_i^+\} \tag{2.5}$$

$$U_i(1) = \{v_j | v_j \in N_i^-\} \tag{2.6}$$

$$Bi(l+1) = \{u_j | u_k \in B_i(l), u_j \in N_k^+\} \cup \{u_j | u_k \in U_i(l), u_j \in N_k^-\} \tag{2.7}$$

$$Ui(l+1) = \{u_j | u_k \in U_i(l), u_j \in N_k^+\} \cup \{u_j | u_k \in U_i(l), u_j \in N_k^-\} \tag{2.8}$$

其中 l 为正整数,N_i^+ 和 N_i^+ 分别指节点 i 的正邻居节点集合和负邻居节点集合。

基于扩展的结构平衡理论，就关系的密切性而言，我们有以下结论：正连接的关系比无连接的关系更近，无连接的关系比负连接的关系更近。

2.1.3 向量的相似性度量

向量之间的欧拉距离表示为两个向量 $\mathbf{x}$ 与 $\mathbf{y}$ 差值的模：

$$d_E(\mathbf{x}, \mathbf{y}) = |\mathbf{x} - \mathbf{y}| \tag{2.9}$$

为了提高运算效率，我们有时会使用欧拉距离的平方也即向量之差的每个分量的平方和来表征，即

$$d'_E(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_2^2 \tag{2.10}$$

对于向量 $\mathbf{x}$ 与 $\mathbf{y}$, 余弦相似度的数学表达式为：

$$sim(x,y) = \cos(\theta) = \frac{\mathbf{x} \cdot \mathbf{y}}{|\mathbf{x}||\mathbf{y}|} \tag{2.11}$$

其中，θ 是向量 $\mathbf{x}$ 和 $\mathbf{y}$ 之间的夹角，$|\mathbf{x}|$ 表示 $\mathbf{x}$ 的模长。

节点间的余弦相似度可以用来衡量两个节点嵌入表示之间的相似程度，其取值范围在 $[-1,1]$ 之间，其中 1 表示两个向量完全相同，-1 表示两个向量完全不同，而 0 则表示两个向量之间没有相关性。为了将余弦相似度转化为一个非负且变化相对更加明显的量，我们对其取指数得到：

$$S_{ij} = e^{sim(i,j)} \tag{2.12}$$

2.2 社团划分

2.2.1 社团划分的目的

社团划分问题 (community detection)，又称社区检测问题，是指将一个网络中的节点划分为几个社团，使得社团内部的节点之间的连接比较紧密，社团外部的节点之间连边稀疏。对于符号网络上的社团划分问题，我们希望社团内部节点连接紧密的同时，有尽可能多的连边为正边，社团之间节点连接稀疏的同时，尽可能多的连边为负边。从社会网络的现实意义上来考虑，符号网络的社团划分相当于是将网络划分为几个小团体，小团体之间关系对立，小团体内部成员之间关系比较团结。

2.2.2 评价标准

本节中我们介绍社团划分的两种评价标准：模块度、NMI（标准化互信息）。模块度[6]描述了一个网络或图像中节点之间的紧密程度和组织方式，可以用来衡量网络社团划分的效果。模块度高的网络或图像意味着节点之间的连接更紧密、组织更有序，分组更加合理，而模块度低的网络或图像则意味着节点间连接更松散、组织更混乱，分组不够合理。

对于符号网络，模块度[7]定义的表达式可以表示为：

$$Q = \frac{1}{2m^+ + 2m^-} \sum_{i,j} [a_{ij} - \left(\frac{k_i^+ k_j^+}{2m^+} - \frac{k_i^- k_j^-}{2m^-} \right)] \delta(c_i, c_j) \tag{2.13}$$

其中，m^+ 与 m^- 分别指的是正边总数与负边总数，k_i^+ 与 k_i^- 分别指节点 v_i 的正度与负度（见2.2），$\delta(c_i, c_j)$ 的定义如下：

$$\delta(c_i, c_j) = \begin{cases} 1, & \text{节点 } v_i \text{ 和节点 } v_j \text{ 属于同一个社团} \\ 0, & \text{节点 } v_i \text{ 和节点 } v_j \text{ 属于不同社团} \end{cases} \tag{2.14}$$

正如前面所提到的，社团划分问题可以看作节点聚类问题的一种特殊情况，故而，衡量聚类结果与真实结果相似度的标准化互信息（NMI, Normalized Mutual Information）[8]，作为聚类问题的重要指标，也可以用作评价社团划分的结果。设 X 和 Y 是两个聚类结果，其联合分布为 $P(X,Y)$，边缘分布分别为 $P(X)$ 和 $P(Y)$，则互信息 $I(X;Y)$ 可以定义为：

$$I(X;Y) = \sum_{x\in X}\sum_{y\in Y} P(x,y)\log\frac{P(x,y)}{P(x)P(y)} \tag{2.15}$$

其中，$\sum_{x\in X}\ \sum_{y\in Y}$ 表示对所有可能的 x 和 y 进行求和，$P(x,y)$ 表示 $X=x$ 且 $Y=y$ 的概率，$P(x)$ 和 $P(y)$ 分别表示 $X=x$ 和 $Y=y$ 的边缘概率，log 是自然对数。

为了消除样本个数等的影响，我们将互信息 $I(X;Y)$ 标准化，也即将其除以 X 和 Y 的熵之和，得到标准化互信息的定义：

$$NMI(X,Y) = \frac{2I(X;Y)}{H(X)+H(Y)} \tag{2.16}$$

NMI 的取值范围为 $[0,1]$，值越大表示两个聚类结果越相似，当 NMI 等于 1 时，表示两个聚类结果完全相同。我们往往用我们所得到的的预测结果与真实划分结果的标准化互信息值来评估社团划分的结果。

第三章 图卷积神经网络

3.1 嵌入表示

我们通过图神经网络，可以将网络中每个节点的结构特征提取出来，作为向量输出，这就是节点的嵌入表示（Node Embedding）。在节点的嵌入表示中，每个节点都被映射为一个向量，这个向量通常具有比原始图数据（图网络的邻接矩阵或者拉普拉斯矩阵）更低的维度，这使得它们更容易处理和可视化。因此，我们可以说，节点的嵌入表示是将节点映射到低维向量空间的一种技术。

节点的嵌入表示可以通过各种算法来实现，其中包括基于深度学习的方法和传统的线性代数方法，比如主成分分析（PCA）。本文接下来要介绍的算法里，所生成的节点嵌入表示，就是通过一种图卷积神经网络，也就是深度学习方法得出的。

节点的嵌入表示作为图网络数据的一种低维表征，可用于各种图网络分析任务，比如节点聚类，链接预测等等。

3.2 聚合过程

本文中主要涉及的都是空域上的图卷积神经网络。空域上的图卷积神经网络的主要思想就是对图网络中，当前节点与其邻居节点的信息进行聚合，根据所聚合的信息生成节点的嵌入表示。

那么具体来讲，这个聚合过程是如何进行的呢？我们将以 GCN 的一个变体——GraphSAGE[9] 为例展开介绍。

GraphSAGE 采用的是应对于大型图的小批量训练方法，也即每次从图中选取一小批节点进行训练。本文中为方便说明起见，直接对图上的每个节点展开聚合。

我们的输入包括：图 $G = (V, B)$，节点数为 n；输入特征 $X = \{x_v, \forall v \in V\}$，记为 $n \times d^{in}$；层数 K；激活函数 σ; 邻居节点聚合函数 $Agg^{(k)}, \forall k \in \{1, ..., K\}$；权重矩阵 $W^{(k)}, \forall k \in \{1, ..., K\}$；邻居采样函数 $N^{(k)} : v \to 2^v, \forall k \in \{1, ..., K\}$（此处采样是指对每个节点，在每层固定最多取 S_k 的邻居节点, 且邻居节点被选中的概率是相等的）。

$h_u^{(k)}$ 表示第 k 层的隐藏层输出，比如节点嵌入表示学习，输出的就是节点的嵌入表示，其维数是由我们预先指定的，我们将其记为 d^{out}。

具体流程如下：

- 设定初始的输入，$h_u^{(0)} \leftarrow X$;

- 我们遍历每一层，对于第 k 层（$k = 1, 2, ..., K$）
  - 对于 V 中每个节点 v，我们有：
    - $h_{N(v)}^{(k)} \leftarrow Agg^{(k)}(\{h_u^{(k-1)}, \forall u \in N^{(k)}(v)\})$
    - $h_v^{(k)} \leftarrow \sigma(W^{(k)}[h_v^{(k-1)}||h_{N(v)}^{(k)}])$
    - $h_v^{(k)} \leftarrow h_v^{(k)}/||h_v^k||_2$
- 遍历结束之后，我们将最后一层的结果输出：$z_v \leftarrow h_v^{(k)}, \forall v \in V$

其中，$Agg^{(k)}$ 作为邻居节点的聚合算子，有不止一种形式：平均聚合算子，加和聚合算子，最大池化聚合算子等。我们以平均聚合算子为例来介绍一下邻居节点是如何聚合的。平均聚合算子实际上就是逐元素取均值，对于节点 v 而言，其平均聚合算子的形式为：

$$Agg^{mean} = AVERAGE(h_u, \forall u \in N(v)) \tag{3.1}$$

于是，对节点 v 的邻居采样节点 u 对应 $h^{(k-1)}$ 取均值，便得到了 $h_{N(v)}^k$。

当层数 $k = 1$ 时，我们节点 v 的输入 $h_v^{(0)}$ 为 d^{in} 维的，我们所得到的 $h_{N(v)}^{(1)}$ 的向量也是 d^{in} 维的，我们将这两个向量缀连起来，得到一个 $2d^{in}$ 维度的向量 $[h_v^{(0)}||h_{N(v)}^{(1)}]$。

我们用 $d^{out} \times 2d^{in}$ 维的权重矩阵 $W^{(1)}$ 与该向量相乘以进行线性变换，得到一个 d^{out} 维的向量，我们再将该向量的各分量经过一个激活函数进行非线性变换，就完成了第二步聚合操作，得到了 d^{out} 维的向量 $h_v^{(k)}$。

我们再对该向量除以其 L2 范数进行归一化，对节点 v 就完成了第一层的聚合操作。

当层数 $k > 1$ 时，我们的节点 v 的输入 $h_v^{(k-1)}$ 作为上一层的输出，为 d^{out} 维的向量，我们将得到的 $h_{N(v)}^{(k)}$ 的向量与之缀连起来，得到一个 $2d^{out}$ 维度的向量 $[h_v^{(k-1)}||h_{N(v)}^{(k)}]$。

我们用权重矩阵 $d^{out} \times 2d^{out}$ 维的权重矩阵 $W^{(k)}$ 与该向量相乘以进行线性变换，对得到的 d^{out} 维的向量经过一个激活函数进行非线性变换，就完成了第二步聚合操作，得到了 d^{out} 维的 $h_v^{(k)}$。我们再对该向量除以其 L2 范数进行归一化，对节点 v 就完成了第 k 层的聚合操作。

由于 GraphSAGE 是基于邻居聚合的方法，因此对于没有直接联系但在同一社交网络或生物网络中紧密相关的节点也能进行有效的表示学习。

3.3 损失函数

为了让节点的嵌入表示更能反映图网络的结构特征，我们需要在训练过程中加入损失函数以更新参数。具体来讲，GraphSAGE 无监督学习的损失函数是一个负采样损失函数，其数学表达式如下：

$$L = -\sum_{i=1}^{N}\{\sum_{j\in\mathcal{N}_i}\log\sigma(\mathbf{z}_i^T\cdot\mathbf{z}_j) + k\cdot\mathbb{E}_{j\sim P_n(i)}[\log\sigma(-\mathbf{z}_i^T\cdot\mathbf{z}_j)]\} \tag{3.2}$$

其中，N 是节点总数，$\mathcal{N}_i$ 是节点 i 的邻居节点集合，k 是负采样数量，$P_n(i)$ 是节点 i 的负采样分布，$\mathbf{z}_i$ 是节点 i 的嵌入向量，σ 是 sigmoid 函数, 作用是将节点 i 与 j 的点乘（相似性表征）转化为 (0,1) 间的概率分布。

该损失函数的目的是使得邻接节点之间的相似性与无连接节点之间的相似性差值尽可能增大。

3.4 普通 GCN

我们来总结一下空域中图卷积神经网络的一般流程。

我们的输入包括：$n\times n$ 的邻接矩阵 A, 节点的特征矩阵 $X(n\times d^{in})$ 维（d^{in} 表示为每个节点输入的特征维数），聚合层数 L，对邻居节点聚合操作，我们简化地表示为函数 f。我们的输出是一个 $n\times d^{out}$ 维的表示矩阵 Z。

- 设定初始的 $H(0)$ 为输入的特征矩阵 X。也就是：

$$H^{(0)} \leftarrow X \tag{3.3}$$

- 我们遍历每一层，对于第 l 层 $(l = 1, 2, ..., L)$，我们有

$$H^{(l)} \leftarrow f(H^{(l-1)}, A) \tag{3.4}$$

- 遍历完成之后，我们根据 $H^{(l)}$ 的损失函数，更新 GCN 的参数。
- 重复以上两步直到损失函数收敛，这时候我们输出第 L 层的 $H^{(L)}$ 为 Z，也即：

$$Z \leftarrow H^{(L)} \tag{3.5}$$

第四章　符号网络上的社团划分算法

本章主要分为两个部分，我们将在这两部分中分别介绍我们提出的两个算法。我们将首先介绍我们建立在 SGCN（Signed Graph Convolutional Network，符号图卷积神经网络）[5] 基础上的节点无监督聚类算法的基本流程，再介绍基于节点相似度的谱聚类算法。

4.1　面向节点聚类的符号图卷积神经网络

在第三章中我们提到，图卷积神经网络可以生成节点的嵌入表示，这个嵌入表示表征了图网络的一些结构信息，而我们可以对节点的嵌入表示进行聚类，生成图网络的社团划分。我们选取了基于符号网络的一种空域的图卷积神经网络：符号图卷积神经网络（Signed Graph Convolutional Network，简称为 SGCN），并在此基础上对其损失函数作了进一步改进，提供了两种损失函数。通过 SGCN 生成节点的嵌入表示后，我们对这些嵌入表示向量利用聚类算法进行聚类，从而得到节点的社团划分。

原 SGCN 算法研究的是这样一个节点嵌入问题：

给定一个符号网络 $G = (V, E+, E-)$，表示为邻接矩阵 $A \in R^{n\times n}$，我们想要为一个节点寻找到一个低维向量，表征该节点在网络中的特征等。也即：

$$F : A \to Z \tag{4.1}$$

也就是说，我们要找到一个变换函数 F，使得输入符号网络的邻接矩阵 A 时，为每个节点得到一个 d 维的向量以表征该节点的特征；也就是说，我们得到一个 $n \times d$ 维的矩阵，记作 Z.

因为节点的嵌入表示表征了图的一些拓扑结构信息，在通过 SGCN 输出节点的嵌入表示之后，我们可以对节点的嵌入表示进行聚类，最终输出节点的社团划分。明确了算法的主要架构之后，我们来具体看一下算法中具体的聚合、参数更新、以及嵌入表示聚类操作应当如何进行。

4.1.1　基于结构平衡理论的聚合

SGCN 对邻居节点的信息聚合与 GraphSAGE 的聚合方式极其类似，可以视为 GraphSAGE 在符号网络上的变体。

为了理解 SGCN 的聚合过程，我们先来明确一下聚合过程的输入和输出。输入：图 $G = (V, E^+, E^-)$ 与节点的特征矩阵 X（$n\times d^{in}$ 维）（d^{in} 表示为每个节点输入的特

征维数，节点数量为 n），聚合层数 L，激活函数 σ。特别地，假如输入的图中，节点没有初始特征，我们用截断奇异值分解（Truncated SVD）[10]方法将输入图的邻接矩阵降维到 d^{in} 维，作为节点的初始特征矩阵 X。

类似一般的 GCN，我们在开始的时候指定 $h_i(0) \in R^{d^{in}}$ 为节点 i 初始的特征，也即:

$$h_i^{(0)} \leftarrow x_i, \forall i \in V$$

来表示初始输入的节点 i 的特征。

输出: 一个 $n \times 2d^{out}$ 维的嵌入表示矩阵 Z，其第 i 行表示节点 i 的嵌入表示，形式为:

$$z_i \leftarrow [h_i^{B(L)} || h_i^{U(L)}]$$

其中 L 指的是 SGCN 的总层数。z_i 是由节点 i 的“朋友”嵌入表示 $h_i^{B(L)}$ 与“敌人”嵌入表示 $h_i^{U(L)}$ 连缀而成的向量，节点的“朋友”信息和“敌人”信息是分别聚合的。$B(l)$ 与 $U(l)$ 分别表示第 l 层的“朋友”聚合器与敌人聚合器，关于此聚合器具体是如何操作的，我们会在后面作进一步介绍。

嵌入表示 z_i 的维度同样是由我们事先指定的，记为 $2d^{out}$，也就是说，$h_i^{B(L)}$ 与 $h_i^{U(L)}$ 分别均为 d^{out} 维的。

对于节点 i 而言，其第 l 层的“朋友”聚合信息来源于以下这些节点的 $l-1$ 层信息:

- 节点 i;
- 节点 i 的 l 阶平衡集（见2.1.2中对平衡集的定义）: $B_i^{(l)}$，它包括两部分:
  - 节点 i 的正邻居节点的 $l-1$ 阶平衡集;
  - 节点 i 的负邻居节点的 $l-1$ 阶非平衡集。

对于其“敌人”聚合信息的聚合则反之，聚合自节点 i 与 i 的 l 阶非平衡集的节点的信息。$l = 1, 2, 3 \ldots\ldots L$ 表示与 u_i 相连的路径的长度，也表示在 SGCN 的第几层。

我们认为节点 i 的平衡集中的节点，可以认为是 i 的“朋友”；而 i 的不平衡集中的节点，我们推测是 i 的“敌人”。

我们可以看到节点 i 的两个正邻居，也就是 $B_i^{(1)}$ 的信息，将通过使用“朋友”聚合器 $B(1)$ 被合并到第一层的“朋友”信息中。也就是说，对于层数 l 而言，$B(l)$ 表示的是聚合当前节点 i 与 $B_i(l)$ 中的元素的信息。

对于节点 i 而言，第一层的聚合公式为:

$$h_i^{B(1)} = \sigma(W^{B(1)}[\sum_{j \in N_i^+} \frac{h_k^{(0)}}{|N_i^+|} || h_i^{(0)}]) \tag{4.2}$$

$$h_i^{U(1)} = \sigma(W^{U(1)}[\sum_{j \in N_i^-} \frac{h_k^{(0)}}{|N_i^-|} || h_i^{(0)}]) \tag{4.3}$$

我们以 $h_i^{B(1)}$ 为例进行说明, 其中 $h_i^{(0)}$ 表示节点 i 的初始输入特征，$W^{B(1)}$ 是一个 $d^{out} \times 2d^{in}$ 的权重矩阵，σ 表示激活函数。

我们将节点 i 的正邻居节点 j 对应的 $h_j^{(0)}$ 取平均后得到的向量，与节点 i 对应的 $h_i^{(0)}$ 串联起来之后，得到的 $[\sum_{j \in N_i^+} \frac{h_k^{(0)}}{|N_i^+|} || h_i^{(0)}]$ 是一个 $2d^{in} \times 1$ 的向量。我们将权重矩阵 $W^{B(1)}$ 与该向量相乘，作一步线性变换。因为我们想要输出为 $2d^{out}$ 维的嵌入向量，我们的权重矩阵为 $d^{out} \times 2d^{in}$ 维度的，如此，两者相乘所得到的的向量即为 d^{out} 维度的。

经过一步线性变换之后，我们再对该向量的每一个分量经过一个激活函数以进行非线性变换，就得到了节点 i 的 1 层“朋友”表示：$h_i^{B(1)}$。

对 $h_i^{U(1)}$ 的计算也是类似的，需要注意的一点是，这里的 $W^{B(1)}$ 与 $W^{U(1)}$ 作为节点“朋友”表示与“敌人”表示中的参数，是分别进行更新的。

对于节点 i 而言，第 l 层 $(l \geq 1)$ 的聚合公式为:

$$h_i^{B(l)} = \sigma(W^{B(l)}[\sum_{j \in N_i^+} \frac{h_j^{B(l-1)}}{|N_i^+|} || \sum_{k \in N_i^-} \frac{h_k^{U(l-1)}}{|N_i^-|} || h_i^{B(l-1)}]) \tag{4.4}$$

$$h_i^{U(l)} = \sigma(W^{U(l)}[\sum_{j \in N_i^+} \frac{h_j^{U(l-1)}}{|N_i^+|} || \sum_{k \in N_i^-} \frac{h_k^{B(l-1)}}{|N_i^-|} || h_i^{U(l-1)}]) \tag{4.5}$$

$h_i^{B(l)}$ 这里，我们所聚合的信息包括三部分: 节点 i 的正连接节点的 $l-1$ 层“朋友”表示，节点 i 的负连接节点的 $l-1$ 层“敌人”嵌入表示，与节点 i 的 $l-1$ 层“朋友”嵌入表示, 得到一个 $3d^{out}$ 维的向量 $[\sum_{j \in N_i^+} \frac{h_j^{B(l-1)}}{|N_i^+|} || \sum_{k \in N_i^-} \frac{h_k^{U(l-1)}}{|N_i^-|} || h_i^{B(l-1)}]$。

我们将 $d^{out} \times 3d^{out}$ 维的权重矩阵 $W^{B(l)}$ 与这个 $3d^{out}$ 的向量相乘，得到一个 d^{out} 维度的向量，再经过一个激活函数，就得到节点 i 的 l 层“朋友”表示 $h_i^{B(l)}$。

对节点 i 的 l 层“敌人”表示 $h_i^{U(l)}$ 的计算也是类似的。

4.1.2 损失函数及其改进

我们是想对神经网络生成的嵌入表示进行节点聚类，为了使得节点的嵌入表示更好地反映图中的信息，需要添加损失函数进行参数更新。原 SGCN 的训练目标是：能

够将节点对间的关系分类为正连接，负连接与无连接三类，也即链接符号预测（Link Sign Prediction）。我们的训练目标与原网络的训练目标中不同，于是需要对损失函数作进一步的调整。于是，我们对 SGCN 网络的损失函数进行了重新定义，并因此发现，改进的损失函数所训练出的嵌入表示在链接符号预测问题的表现也得到了提升。

原 SGCN 的损失函数由两部分组成：

$$L_{total} = L_{CE} + \lambda L_{gap} \tag{4.6}$$

其中，λ 是平衡这两项的超参数，L_{CE} 是节点对分类的交叉熵损失函数：

$$L_{CE} = -\frac{1}{|M|} \sum_{(i,j,s)\in M} w_s log \frac{exp(\theta_s[z_i||z_j] + b_s)}{\sum_{q\in\{+,-,?\}} exp(\theta_q[z_i||z_j] + b_q)} \tag{4.7}$$

这里，M 是我们为更新损失函数，从图网络中采样得到的节点对集合，其中 s 表示的是节点对 i 和 j 之间的真实连接情况（正边连接，负边连接或无边连接）。$w_s = \frac{1}{n_s}$（n_s 表示类别的样本点数量），是为了平衡类别中样本点数量的权值。为节点对 i 与 j 的嵌入表示 $[z_i||z_j]$ 这个 $4d^{out}$ 维度的向量，我们添加了一个全连接层，θ_q,b_q 均是该层的权重参数，以将节点对嵌入表示 $[z_i||z_j]$ 的 $4d^{out}$ 维度数据转化为对节点对三类 $(+,-,?)$ 对应的输出，我们再将此输出经过一个 softmax 函数将其转化为一个概率分布。

softmax 函数作为一种归一化函数，可以将一组任意实数转换为一个概率分布，常用于多分类问题，其表达式为：

$$\text{softmax}(z_i) = \frac{exp(z_i)}{\sum_{j=1}^{K} exp(z_j)}, \quad i = 1, \ldots, K \tag{4.8}$$

其中，K 为分类的类别个数，z_i 此处指的是向量 z 的第 i 个分量。分类问题中，对于向量 z 而言,softmax 的函数值也就是 z 属于第 i 类的概率。

所以，这里实际上就是将节点对 i,j 的嵌入表示向量 $[z_i||z_j]$，转化成该节点对分类为 $(+,-,?)$ 中真实类别的概率分布，我们将其求均值后取负，即可以在训练过程中通过最小化损失函数，最大化节点对分类为真实连接情况的概率，于是我们得到了节点对分类的交叉熵损失函数项。

损失函数中的 L_{gap} 项定义为：

$$\begin{aligned} L_{gap} =& \frac{1}{|M(+,?)|} \sum_{(i,j,k)\in M(+,?)} max(0, (||z_i - z_j||_2^2 - ||z_i - z_k||_2^2)) \\ &+ \frac{1}{|M(-,?)|} \sum_{(i,j,k)\in M(-,?)} max(0, (||z_i - z_k||_2^2 - ||z_i - z_j||_2^2)) \end{aligned} \tag{4.9}$$

其中，$M(+,?)$ 表征的是正边相连的节点对集合中，对每一个节点对 (i,j)，我们引入一个未与 i 相连的节点 k(每次迭代都选取不同的 k)，$M_{(-,?)}$ 与之同理；$||z_i - z_j||_2^2$ 表征的是节点 i 与节点 j 之间欧几里得距离的平方。

L_{gap} 是依据扩展的结构平衡理论附加的一项。根据符号网络的正负连边的语义内涵，我们认为：正连接的关系比无连接的关系更近，无连接的关系比负连接间的关系更近。也就是说，正连接节点间的嵌入表示的相似性应当高于无连接节点间的嵌入表示相似性，无连接节点间的嵌入表示相似性应当高于负连接节点间的嵌入表示相似性。这也就是 L_{gap} 的目的。

为了更准确地对节点进行聚类，我们对损失函数作了进一步的修改。

前面我们提到，我们的训练目标是将网络划分为社团，使得社团内部联系紧密且连边尽量为正边，社团之间联系稀疏且连边尽量为负边；表现为节点内部的嵌入表示比较相似，节点之间的嵌入表示差异较大。

因为我们的社团划分是无监督算法，而交叉熵损失函数是基于真实分类展开计算的，于是我们不再保留损失函数中的交叉熵损失函数项。

我们认为，当两节点间有正边连接时，节点间的嵌入表示的相似度应当相对比较高；当两节点间有负边连接时，节点间的嵌入表示相似度应当相对比较低；而对于两节点无连接的状况，我们认为节点相似度应当居于两者之间。

参考 GraphSAGE 对嵌入训练的损失函数表达（见式（3.2）），我们将原损失函数初步更改为：

$$\begin{aligned} L = & \frac{1}{|M(+,?)|} \sum_{(i,j,k)\in M(+,?)} (||z_i - z_j||_2^2 - ||z_i - z_k||_2^2)) \\ & + \frac{1}{|M(-,?)|} \sum_{(i,j,k)\in M(-,?)} (||z_i - z_k||_2^2 - ||z_i - z_j||_2^2)) \end{aligned} \tag{4.10}$$

与原 L_{gap} 项相比，从正连接与无连接对应的第一项来看，我们保留了当正连接节点 i、j 之间的距离小于等于节点 i 与负采样的无连接节点 k 间距离时，两者的距离差值。也就是说，我们要最小化节点对之间的正连接距离与无连接距离的差值，同时最小化节点 i 与节点 j 之间无连接距离与负连接距离的差值。

在这个损失函数式下，即使正连接节点之间的相似性不比无连接时节点间相似性低，我们也依然会最大化正连接节点的相似性，进一步增加其与无连接节点间相似性的差异。换句话说，此损失函数的意义在于最大化正连接节点之间的相似性与无连接节点间相似性的差值，同时最大化无连接节点间相似性与负连接节点对之间相似性的差值。

我们据此发现，对于符号网络而言，实际上，我们要做的就是最大化正连接节点与负连接节点的相似度差异。于是，于是我们在上式中去除负采样节点相关量，将损失函数 L 进一步简化地改为：

$$
\begin{aligned}
L_1 =& \frac{1}{|M(+)|} \sum_{(v_i,v_j)\in M(+)} ||z_i - z_j||_2^2 \\
&- \frac{1}{|M(-)|} \sum_{(v_i,v_j)\in M(-)} ||z_i - z_j||_2^2
\end{aligned}
\tag{4.11}
$$

其中，$M(+)$ 与 $M(-)$ 分别为正边集合与负边集合。这样就得到了我们损失函数的其中一种形式。

除却原 SGCN 网络中的损失函数所用到的嵌入表示之间欧几里得距离的平方之外，嵌入表示相似性的衡量标准还有向量的余弦相似度。

引入节点嵌入表示向量间的余弦相似度 (见式 (2.11)) 替换 $||z_i - z_j||_2^2$，我们将损失函数 L 设为：

$$
\begin{aligned}
L_2 =& -\frac{1}{|M(+)|} \sum_{(v_i,v_j)\in M(+)} S_{ij} \\
&+ \frac{1}{|M(-)|} \sum_{(v_i,v_j)\in M(-)} S_{ij}
\end{aligned}
\tag{4.12}
$$

$S_{ij} = exp[sim(z_i, z_j)]$(见式 (2.12)) 为 z_i 与 z_j 的余弦相似度取指数。这里的损失函数意义也是最大化节点对正连接相似度与负连接相似度的差值。这就是我们所提出的另一种损失函数形式。实际应用时，我们可以根据实际情况设定节点的相似性表征。

我们根据上面的损失函数，利用随机梯度下降算法 (SGD)[11] 更新参数，输出节点的嵌入表示。

4.1.3 嵌入表示的聚类

我们对生成的节点嵌入表示进一步进行聚类, 所输出的就是节点的社团划分。此时，我们已经把社团划分问题转化为了一个对节点的嵌入表示，也就是向量进行聚类的问题。而业界对于这一问题的研究是相对比较成熟的，有许多效果很好的算法，比如 K-Means，层次聚类算法等等。

我们将图网络数据转化为节点的嵌入表示之后，会运用聚类算法对嵌入表示向量进行聚类。

聚类算法是一种无监督学习，其目的是将数据集中的样本划分为若干个类别，使得同一类别内的样本相似度相对比较高，而不同类别间的样本相似度相对比较低。聚类算法可以应用于许多领域，如图像处理、文本挖掘、生物信息学等。

考虑到社团划分问题，我们很容易想到，倘若我们通过计算出节点间的相似度或距离，或者说通过深度学习等方法提取出节点的结构特征，那么我们就可以通过对这些节点数据进行聚类，将网络划分为内部紧密联系而外部联系相对松散的社团。事实上，节点聚类算法在社团划分问题中是比较常用的。

常见的聚类算法包括 K-Means、层次聚类、DBSCAN、谱聚类等。其中，K-Means 和层次聚类是最常用的聚类算法之一。下面我们将分别简要叙述一下 K-Means, 层次聚类的基本原理与各自的优缺点。

K-Means[12] 算法通过迭代更新簇中心来进行聚类:

1. 首先随机选择 k 个样本作为初始簇中心，然后将剩余的样本划分到距离最近的簇中心所在的簇中。

2. 接着重新计算每个簇的中心，并再次将所有样本划分到距离最近的簇中心所在的簇中。

3. 重复以上步骤直到簇中心不再发生变化或达到预设的最大迭代次数。

K-Means 算法的优点是速度快，对于大规模的数据集有较好的表现，但需要预先指定簇的个数 K, 对于数据集的分布和密度要求较高，对于噪声和异常值比较敏感。

层次聚类算法可以分为自底向上的凝聚层次聚类和自顶向下的分裂层次聚类两种方式。凝聚层次聚类 (Agglomerative Clutsering)[13] 首先将每个样本看作一个独立的簇，然后将距离最近的两个簇合并成为一个新的簇，不断重复该过程直到所有样本都被合并为同一个簇。分裂层次聚类（Divisive Clstering）则是从一个包含所有样本的簇开始，递归地将其划分为多个子簇，直到每个簇只包含一个样本。我们一般使用自底而上的凝聚层次聚类。

层次聚类算法的优点是可以自动选择簇的个数，对不同密度和分布的数据集都有较好的表现，但是计算复杂度高，对于大规模数据集的计算时间较长。

本文中，我们对输出的嵌入表示采用 K-Means 与自底而上的凝聚层次聚类算法（Agglomerative Clustering）两种方式进行聚类。

4.1.4 算法流程

我们通过改进的 SGCN 模型进行社团划分的总体流程如下:

- 对每个节点 $i \in V$，设定初始的 $h_i^{(0)}$ 为输入的特征矩阵 X。也就是:

$$h_i^{(0)} \leftarrow x_i \tag{4.13}$$

- 对于每个节点 $i \in V$，依照式 (4.2) 和式 (4.3) 进行第一层聚合，得到 $h_i^{B(1)}$,$h_i^{U(1)}$
- 假如层数 L 大于 1，我们遍历每一层，对于第 l 层 $(l = 2, 3, ..., L)$：
  - 对于每个节点 $i \in V$，我们根据式（4.4）与式（4.5）进行第 l 层聚合，得到 $h_i^{B(l)}$ 与 $h_i^{U(l)}$
- 第 L 层完成聚合之后，我们得到

$$z_i \leftarrow [h_i^{B(L)}||h_i^{U(L)}] \tag{4.14}$$

- 本轮遍历完成之后，我们根据损失函数（如（式4.12）或式（4.11）），更新 SGCN 的参数。
- 重复进行前向传播与更新参数，直到损失函数收敛，这时候我们以此时的第 L 层聚合结果 z_i（见式（4.14））作为节点的嵌入表示输出。
- 将输出的节点嵌入表示 z_i 通过 K-Means 或层次聚类算法进行聚类，得到网络的社团划分。

4.2 基于节点相似度的谱聚类

除却基于改进的 SGCN 模型进行节点聚类之外，我们还提出了另一种社团划分算法，其基本原理为基于符号网络中的节点相似度，得出网络的相似度矩阵，并以此进行谱聚类，从而得到节点的社团划分。

4.2.1 节点相似度

对于符号网络，节点 v_i 与 v_j 之间的相似度[3]定义为：

$$s^+(v_i, v_j) = |\Gamma^+(v_i) \cap \Gamma^+(v_j)| + |\Gamma^-(v_i) \cap \Gamma^-(v_j)| \tag{4.15}$$

$$s^-(v_i, v_j) = |\Gamma^+(v_i) \cap \Gamma^-(v_j)| + |\Gamma^-(v_i) \cap \Gamma^+(v_j)| \tag{4.16}$$

其中，$\Gamma^+(v_i)$ 和 $\Gamma^-(v_i)$ 分别指节点 v_i 的正边邻居节点集合与负边邻居节点集合，

$$s_{ij} = \frac{s^+(v_i, v_j) - s^-(v_i, v_j) + a_{ij}}{\{\Gamma(v_i), v_i\} \cup \{\Gamma(v_i), v_i\}} \tag{4.17}$$

由此，我们可以定义该网络的节点相似度矩阵 S，其中第 i 行 j 列的元素为其 i 与 j 节点之间的余弦相似度取指数：

$$S_{ij} = e^{s_{ij}} \tag{4.18}$$

4.2.2 拉普拉斯矩阵

图的拉普拉斯矩阵是一种表示图结构的数学工具，通常用 L 表示。对于一个 n 个节点的无向图 $G = (V, E)$，其拉普拉斯矩阵 L 的定义为：

$$L = D - A \tag{4.19}$$

其中 A 是 $n \times n$ 的邻接矩阵，D 是 $n \times n$ 的度矩阵，$D_{i,i}$ 表示节点 i 的度数（即与之相连的边的条数），其它元素为 0。

拉普拉斯矩阵有很多重要的性质，比如它是对称半正定矩阵，其最小特征值为 0，且对应的特征向量为全 1 向量。这些性质使得拉普拉斯矩阵在图论、谱聚类、图像分割等领域有广泛的应用。

因为在符号网络中，节点之间的连边有正负之分，不适合用谱聚类算法直接进行划分，所以我们用节点之间的相似度矩阵构建新的度矩阵，从而构建新的拉普拉斯矩阵。我们知道度矩阵是对角矩阵。对于一个含有 n 个节点的符号网络的相似度矩阵 S（如式（4.18）），我们得出其对应度矩阵 D 第 i 行 i 列的元素为：

$$d_i = \sum_{j=1}^{n} S_{ij} \tag{4.20}$$

其中 S_{ij} 为节点的相似度矩阵 i 行 j 列对应元素，也即第 i 条数据与第 j 条数据之间的相似度。

此时的拉普拉斯矩阵为：

$$L = D - S \tag{4.21}$$

4.2.3 算法流程

谱聚类算法[14]作为一种基于图论的聚类方法，对于高维非线性和噪声数据的处理有着比较优异的表现，同时有着较高的可扩展性和灵活性。

我们根据节点相似度（见式（4.17））构建节点相似度矩阵，再根据谱聚类算法进行节点聚类，据此进行社团划分。对于一个 n 个节点的符号网络，具体的流程如下：

1. 计算拉普拉斯矩阵：据前述内容，计算得到符号网络的相似度矩阵 S，并根据矩阵 S 计算得到拉普拉斯矩阵 L。

2. 特征分解：通过对拉普拉斯矩阵 L 进行特征向量分解，得到一系列特征向量和对应的特征值。

3. 选取特征向量。将 n 个特征向量根据对应特征值从小到大进行排序，并取前 k 个特征向量作为新的表示，其中 k 是聚类数目，对得到的特征向量除以其 L2 范数进行归一化处理，最终得到一个 $n \times k$ 的矩阵 X。

4. 聚类：对得到的矩阵 X 每一行作为一个 k 维的样本，对这 n 个样本进行聚类，通常使用 K-Means 等聚类算法来完成。

第五章 实验

我们完成了改进符号图神经网络与基于节点相似度的谱聚类这两种社团划分算法的编程实现，在本章中，我们将给出其实验结果。我们在人工生成数据集与实际网络上分别进行了实验，采用模块度与 NMI（标准化互信息）作为评价指标。

5.1 数据集

我们依照符号随机块模型[15]生成有社团结构的人工数据集的具体流程如下：

- 设定节点数 N, 由此构成节点序列 0,1,2,3,...,N-1。
- 设定社团划分。比如，我们将网络设定为三个社团，设定 $[0, \frac{N}{3})$ 中的节点为第一个社团，$[\frac{N}{3}, \frac{2N}{3})$ 中的节点为第二个社团，$[\frac{2N}{3}, N)$ 中的节点为第三个社团。
- 为节点间分配连边。我们设定社团内部节点有连边的概率为 p_{in}, 这些连边为正的概率为 p_{in}^{+}; 我们设定社团之间的节点有连边的概率为 p_{out}, 这些连边为正的概率为 p_{out}^{+}；具体来讲，我们遍历图中的节点对，对节点序列中的节点对 i,j：
  - 如果 i,j 是属于同一个社团，i 与 j 之间存在连边的概率为 p_{in}；该连边为正的概率为 p_{in}^{+}, 为负的概率自然为 $p_{in}^{-} = 1 - p_{in}^{+}$；
  - 如果 i,j 是属于不同社团，i 与 j 之间存在连边的概率为 p_{out}；该连边为正的概率为 p_{out}^{+}, 为负的概率为 $p_{out}^{-} = 1 - p_{out}^{+}$。

我们默认设定 p_{in} 为 0.5，p_{out} 为 0.1，p_{in}^{+} 与 p_{out}^{+} 分别为 0.8 与 0.2.

除了上述人工数据集之外，我们还采用了美国国会选票网络 (CV), 修道院网络 (MN)，一战国家关系网络 (War-I)。表5.1给出了我们所采用的数据的相关信息。

表 5.1: 实验网络的统计信息

网络名称	节点数量	连边总数	正边数量	社团数量
N3-300	300	10558	6614	3
N3-500	500	10365	6503	3
War-I	15	105	44	2
MN	18	110	56	3
CV	219	519	413	未知

5.2 划分结果

我们首先研究了 SGCN 的损失函数在去除负采样节点之后，节点嵌入表示在原网络所研究的链接预测问题上的表现。我们将原损失函数 $L_{total} = L_{CE} + \lambda L_{gap}$ 中的 L_{gap} 项更换为我们的第一种损失函数形式（见式（4.11））, 设定 λ 为 5。在同为 100 个 epoch 的情况下，对于 N3-300，改进损失函数前后的 AUC(ROC 曲线下面积) 分别为 0.705，0.791；对于 N5-500，改进前后的 AUC 分别为 0.686，0.752。

于是我们可以初步得出结论，去除负采样节点之后，所输出的嵌入表示能够更充分地反映图网络的结构信息。

我们要验证的目标是：我们对损失函数进行改进之后，生成的节点嵌入表示在聚类中的表现是否得到了提升？为了研究这个问题，我们在有三个社团，300 个节点和 500 个节点（分别表示为 N3-300,N3-500）的人工生成网络上，分别用原 SGCN 与改进后的 SGCN（我们记为 SGCN-C）进行训练，对得到的嵌入表示分别进行聚类。

我们通过实验发现，对 SGCN-C 的嵌入表示进行层次聚类与进行 K-Means 聚类所得到的社团划分结果差异不大，K-Means 算法作为计算复杂度相对较低的算法，在大型网络的处理中更为适用。本文中，我们只列出对嵌入表示采用层次聚类算法进行聚类的划分结果，下面的描述中，对嵌入表示进行聚类，所指的即是层次聚类算法。

我们将利用第一种损失函数形式 (欧几里得距离的平方表征嵌入表示相似性，见式（4.11）) 输出嵌入表示进行聚类的方法记作 SGCN-C1，对利用我们所提出的第二种损失函数 (余弦相似度表征嵌入表示相似性，见式（4.12）) 输出嵌入表示进行聚类的方法记为 SGCN-C2，将基于节点相似度的谱聚类算法记为 Spectral。作为对照组，我们记对原网络的嵌入表示进行层次聚类的结果为“原 SGCN”（设定参数 λ 为 5)。

我们得到原 SGCN 与改进的三种算法的社团划分，与真实社团划分的 NMI 值（见式（2.16））结果如表5.2。

表 5.2: 原网络与改进算法的 NMI 对比

data	原 SGCN	SGCN-C1	SGCN-C2	Spectral
N3-300	0.6983	1.0	1.0	1.0
N3-500	0.5401	1.0	1.0	1.0

由表5.2我们可以看出，我们改进了符号图神经网络的损失函数之后，所得出嵌入表示用于节点聚类，结果水平相较于原本的 SGCN 网络，显著得到了提升。从结果中我们可以看出，对于 N3-300 与 N5-500 这样的人工网络而言，用我们所提出的算法进行社团划分，基本上可以达到比较高的准确率。

对 N3-300 与 N3-500 而言，我们所提出的 SGCN-C 与谱聚类算法得到的社团划分结果，与真实划分的 NMI 值均已经达到了 1.0，得出的符号模块度对应分别均为

0.3436，0.3470。在其他网络数据上，我们的算法进行社团划分的符号模块度结果如表5.3。

表 5.3: 社团划分的符号模块度对比

data	原 SGCN	SGCN-C1	SGCN-C2	Spectral
War-I	0.2269	0.2803	0.3153	0.3153
MN	0.1504	0.1801	0.2499	0.2665
CV	0.1630	0.4395	0.4586	0.4483

由表5.3我们可以看出，我们所提出的两种算法的社团结果均优于原本的 SGCN 网络的嵌入表示直接社团划分，同时 SGCN-C2 的结果普遍优于 SGCN-C1 的结果。于是，我们发现，以余弦相似度来表征节点的嵌入表示的相似性作为损失函数（我们所提出的第二种损失函数，见式 (4.12)），相比于欧几里得距离的平方作为相似性表征，输出的嵌入表示在节点聚类中的表现更加优异。

基于节点相似度的谱聚类在小型图的符号模块度上的表现要略微好于符号图神经网络，结果表现也相对稳定，但该算法的时间复杂度较高，不适用于大型网络。

第六章 总结与展望

6.1 总结

近些年来，社交网络的研究备受关注，本文主要研究的是符号网络上的社团划分问题。而图神经网络作为近年来方兴未艾的一个研究方向，在社团划分等问题研究上也有着独特的优势与应用前景。

本文的主要研究成果有: 回顾了符号网络上已有的社团划分算法，提出了两种符号网络上的社团划分算法:

其一是基于图卷积神经网络的节点聚类方法。我们对符号图卷积神经网络(SGCN)进行改进，修改了其损失函数，以优化其嵌入表示在节点聚类中的表现。我们发现，基于符号网络正连接节点之间的关系近于无连接，而无连接之间的关系又近于负连接的特性，对 SGCN 网络而言，我们去除负采样的无连接节点作为中间量，直接最大化正连接节点与负连接节点之间的表示相似性差异，模型的效果得到了比较明显的提升。

我们在此基础上，将原网络损失函数中衡量向量相似性的欧几里得距离指标替换为向量之间的余弦相似度，提出了一种新的损失函数形式。通过改进的符号图卷积神经网络得到节点的嵌入表示之后，我们对节点的嵌入表示利用 K-Means（或层次聚类算法）进行聚类，得到节点的社团划分。

其二是基于节点相似度的谱聚类算法，我们基于符号网络上的节点相似度构建相似度矩阵，并根据此相似度矩阵进行谱聚类。我们所提出的算法均在各网络数据上取得了不错的结果。

在算法实验中，我们首先验证了我们改进损失后对符号图卷积神经网络模型效果的提升作用，之后我们将所提出的两种社团划分算法在人工生成的符号网络与现实网络的数据集上分别进行社团划分，对划分结果从模块度与标准化互信息两方面分别进行评估，验证了算法对于符号网络社团划分的有效性和准确性。

6.2 展望

目前来看，我们基于已有的成果，接下来可以探讨的研究方向有：(1) 对改进符号图卷积神经网络的损失函数进行进一步的优化。我们将损失函数修改之后，模型在节点聚类上的性能表现提高了，我们将在接下来的工作中探讨其是否仍有进一步提升的空间。

（2）实现改进符号图卷积神经网络的端对端学习。端对端学习作为一种机器学习方法，指的是将输入直接映射到输出。我们目前的算法流程是先通过神经网络生成节点的嵌入表示，再将这些嵌入表示向量输入 K-Means 等算法进行无监督聚类。我们可以在神经网络输出节点的嵌入表示之后，为其添加一个或几个用作节点聚类的层，从而使得神经网络直接输出节点的所属类别。

（3）对于该符号图神经网络，我们将进一步探讨其在图网络的其他问题上该如何扩展，比如预测，异常检测，节点分类问题等等。同时，符号网络作为社交网络的一个重要分支，会涉及到一些大型的图网络，我们也应对算法在更大规模数据上的性能表现进行探讨。

（4）我们的社团划分算法是基于无权无向的符号图的，同时需要预先指定社团数量，而且所划分的社团之间是不存在交集的。接下来，我们将对于自动确定社团数量，重叠社区发现问题，动态社区发现问题进行进一步的研究，如果可能的话，将会以我们的算法在这些问题上进行延展。

参考文献

[1] YANG B, CHEUNG W, LIU J. Community mining from signed social networks[J]. IEEE Trans. on Knowl. and Data Eng., 2007: 1333–1348.

[2] 陈怡然. 符号网络中的社区发现算法研究[D]. 山东大学, 2021.

[3] 甄肖聪. 符号网络中的社团结构研究[J]. 山东大学, 2022.

[4] F.HEIDER. Attitudes and cognitive organization[J]. The Journal of psychology, 1946, 21(1): 107-112.

[5] DERR T, MA Y, TANG J. Signed graph convolutional network: abs/1808.06354[Z]. 2018.

[6] NEWMAN M E J. Modularity and community structure in networks[J]. Proceedings of the National Academy of Sciences, 2006, 103(23): 8577-8582.

[7] GÓMEZ S, JENSEN P, ARENAS A. Analysis of community structure in networks of correlated data[J]. Physical Review E, 2009, 80(1).

[8] DANON L, DÍAZ-GUILERA A, DUCH J, et al. Comparing community structure identification [J]. Journal of Statistical Mechanics: Theory and Experiment, 2005, 2005(09): P09008-P09008.

[9] HAMILTON W L, YING R, LESKOVEC J. Inductive representation learning on large graphs [Z]. 2018. arXiv: 1706.02216.

[10] HALKO N, MARTINSSON P G, TROPP J A. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions[Z]. 2010. arXiv: 0909.4061.

[11] HAMILTON W L, YING R, LESKOVEC J. Inductive representation learning on large graphs [Z]. 2018. arXiv: 1706.02216.

[12] JIN X, HAN J. K-means clustering[M]. Springer US, 2010: 563-564.

[13] ACKERMANN M R, BLöMER J, KUNTZE D, et al. Analysis of agglomerative clustering[J]. Algorithmica, 2012, 69(1): 184-215.

[14] VON LUXBURG U. A tutorial on spectral clustering[Z]. 2007. arXiv: 0711.0189.

[15] YANG B, LIU X, LI Y, et al. Stochastic blockmodeling and variational bayes learning for signed network analysis[J]. IEEE Transactions on Knowledge and Data Engineering, 2017, 29(9): 2026-2039.

致　谢

四年的大学生活即将结束，回顾四年的时光，我得到了身边许多朋友的帮助，在此，我谨向所有曾经关心、支持和爱护我的人们表达我最诚挚的感谢和最美好的祝愿。

感谢我的指导老师亓兴勤老师，她在我的毕业课题研究上给予了很大的帮助，是她的指导帮助我度过了课题研究的瓶颈，对于我论文撰写过程中遇到的问题，她给予了全面悉心的指导意见，同时以言传身教的形式，使我进一步体会到了严谨细致的治学态度的重要性。

感谢我的同班同学，尤其是陪伴我大学时光的舍友们。是你们陪伴我走过大学的四年时光，我感到无比幸运。

最后感谢我的父母，是他们不断支持我的学业与生活，让我没有后顾之忧，他们无微不至的关爱和教诲是我漫漫求学路上最强大的精神动力。

感谢所有给予我帮助，关心、支持和爱护我的人们。