

密级： 保密期限：

# 北京郵電大學

## 硕士学位论文



题目： 动态异质信息网络表示学习研究

学 号： 2017140483

姓 名： 陆元福

专 业： 计算机技术

导 师： 石川

学 院： 计算机学院

2020 年 5 月 15 日



**BEIJING UNIVERSITY OF  
POSTS AND  
TELECOMMUNICATIONS**

# **Thesis for Master Degree**

**Title: Dynamic Heterogeneous Information  
Network Representation Learning**

**Student No.: 2017140483**

**Candidate: Lu Yuanfu**

**Major: Computer Technology**

**Supervisor: Shi Chuan**

**Institute: School of Computer Science**

**May 15, 2020**

### 独创性（或创新性）声明

本人声明所呈交的论文是本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京邮电大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

申请学位论文与资料若有不实之处，本人承担一切相关责任。

本人签名: 陆元福 日期: 2020.5.15

### 关于论文使用授权的说明

本人完全了解并同意北京邮电大学有关保留、使用学位论文的规定，即：北京邮电大学拥有以下关于学位论文的无偿使用权，其中包括：学校有权保留并向国家有关部门或机构送交学位论文，有权允许学位论文被查阅和借阅；学校可以公布学位论文的全部或部分内容，有权允许采用影印、缩印或其它复制手段保存、汇编学位论文，将学位论文的全部或部分内容编入有关数据库进行检索。（保密的学位论文在解密后遵守此规定）

本人签名: 陆元福 日期: 2020.5.15

导师签名: 孙伟 日期: 2020.5.15

# 动态异质信息网的表示学习研究

## 摘要

网络表示学习的快速发展为网络数据分析提供了新颖的研究思路，有效提升了数据挖掘任务的表现。传统网络表示学习方法大多聚焦于静态、同质的信息网络，即假设网络是静止不变、节点和边类型单一。然而，真实世界的网络往往是动态变化、节点和边类型多样的，即所谓的动态、异质信息网络，这使得传统方法无法有效建模网络的真实特性，学到的节点向量表示很难适用于真实场景。因此，本文立足于研究动态、异质的信息网络表示学习算法。具体而言，本文首先研究了网络的动态性问题，并提出结合微观和宏观动态性的动态网络表示学习算法 **M<sup>2</sup>DNE**。其次，本文研究了网络的异质性问题，并提出关系结构感知的异质信息网络表示学习算法 **RHINE**。然后，本文研究了网络的动态性和异质性问题，并提出了动态异质信息网络表示学习算法 **DyHNE**。最后，本文在微信“看一看”的文章推荐场景下，研究了好友增强的推荐问题，并提出社交影响力专注的神经网络 **SIAN**，以验证异质信息网络表示学习在真实场景下的有效性。

真实世界中，动态网络是非常常见的，其通常随着时间在微观和宏观层面不断演化。微观动态性细致地描述了网络结构的形成过程，而宏观动态性表示网络规模的演变模式，两种动态性都是刻画网络演变的重要因素。但是，如何准确地在网络表示学习中捕获两者，尤其是宏观动态性，还没有被很好的研究。因此，为了研究网络的动态性问题，本论文提出了结合微观和宏观动态性的动态网络表示学习算法（**Temporal Network Embedding with Micro- and Macro-Dynamics**，简称 **M<sup>2</sup>DNE**）。对于微观动态性，M<sup>2</sup>DNE 构建时序注意力点过程来捕获网络结构的形成过程。对于宏观动态性，M<sup>2</sup>DNE 定义由网络嵌入表示参数化的的动力学方程。时序动态网络中，微观动态性和宏观动态性的交互的演变交替地影响节点表示的学习过程。在三个真实的动态网络上，丰富的实验验证 M<sup>2</sup>DNE 显著地优于最新方法。

另一方面，信息网络往往是异质的，即网络中包含多类型的节点和边，称作异质信息网络。现有的大多数异质信息网络表示学习方法

通常使用单一模型不加以区分地建模所有关系，这必然会削弱网络嵌入表示的能力。因此，为了研究网络的异质性问题，本论文考虑了异质关系的结构特征并提出了一个新颖的关系结构感知的异质信息网络表示学习模型（**Relation Structure-aware Heterogeneous Information Network Embedding**，简称 **RHINE**）。通过对多个真实异质信息网络数据透彻的数学分析，本论文提出了两种结构相关的度量方法，这两种方法将异质关系一致地划分为两类：隶属关系（Affiliation Relations, ARs）和交互关系（Interaction Relations, IRs）。为了建模关系中有区别的特征，本论文提出两种特别设计的模型分别处理 ARs 和 IRs，以此捕获网络拓扑结构和语义信息。最后，联合各个子模型统一训练优化 RHINE。在四个数据集上，多个任务证明了 RHINE 的有效性。

更进一步地，网络的动态性和异质性是普遍共同存在的，即动态异质信息网络。动态异质信息网络的演变包括多类型节点和边的增加/删除，而一个微小变化会影响整个网络结构和语义信息的变化。为了捕获网络的动态演变，传统异质信息网络表示学习方法需要在每个时间步上重新训练，这是非常耗时且不切实际的。因此，为了研究网络的动态性和异质性，本论文提出了基于元路径相似性的动态异质信息网络表示学习算法（**Dynamic Heterogeneous Information Network Embedding with Meta-path based Proximity**，简称 **DyHNE**）。当网络动态演变时，DyHNE 利用元路径增强的邻接矩阵捕获网络变化。然后通过解决广义特征值问题来学习节点表示，并利用特征值摄动计算节点表示的更新，而不需要重新训练模型。在三个真实动态异质信息网络上，多个任务结果表明模型在有效性和效率性上均优于最新方法。

异质信息网络表示学习的强大建模能力也促进了工业界业务场景的发展。为了验证异质信息网络在真实应用场景中的有效性，本文进一步在微信“看一看”的文章推荐场景下，研究了好友增强的推荐问题，并提出社交影响力专注的神经网络（**Social Influence Attentive Neural Network for Friend-Enhanced Recommendation**，简称 **SIAN**）。为了融合异质信息，SIAN 从异质社交网络的角度建模推荐场景，构建了注意力特征聚合器。更重要的是，SIAN 设计了一个社交影响力耦合器，以捕捉好友推荐圈的影响力。实验结果表明，在两个真实数据集上，所提出的 SIAN 模型优于的最新的些比较方法。

**关键词:**动态网络 异质信息网络 网络表示学习 网络分析 推荐系统

# **DYNAMIC HETEROGENEOUS INFORMATION NETWORK REPRESENTATION LEARNING**

## **ABSTRACT**

The rapid development of network representation learning (a.k.a., network embedding) provides novel ideas for network analysis and effectively improves the performance of data mining tasks. Traditional methods mostly focus on static, homogeneous networks, that is, assuming that the network is static and has a single node and edge type. However, in the real world, networks are usually dynamic, with various node and edge types, so-called dynamic, heterogeneous information networks, which makes traditional methods unable to effectively model the properties of the network. The learned node representations are difficult to apply to real application scenarios. Therefore, this research work is based on studying dynamic and heterogeneous information network embedding methods. Specifically, this paper first studies the dynamics of the network and proposes a dynamic network embedding method M<sup>2</sup>DNE that combines micro and macro dynamics. Secondly, this paper studies the heterogeneity of the network, and proposes a relational structure-aware heterogeneous information network embedding method RHINE. Then, this paper studies the dynamics and heterogeneity of the network, and proposes a dynamic heterogeneous information network embedding method DyHNE. Finally, in the article recommendation scenario of WeChat's "Top Stories", this paper studies the friend-enhanced recommendation and proposes a social influence attentive neural network SIAN to verify the effectiveness of heterogeneous information network embedding in real scenarios.

In the real world, dynamic networks which usually evolve over time in terms of microscopic and macroscopic dynamics, are ubiquitous. The micro dynamics describe the formation process of network structures in a detailed manner, while the macro-dynamics refer to the evolution pattern

of the network scale. Both micro- and macro-dynamics are the key factors to network evolution; however, how to elegantly capture both of them for temporal network embedding, especially macro-dynamics, has not yet been well studied. In this paper, we propose a novel dynamic Temporal Network Embedding with **Micro-** and **Macro-Dynamics**, named **M<sup>2</sup>DNE**. Specifically, for micro-dynamics, we propose a temporal attention point process to capture the formation process of network structures in a fine-grained manner. For macro-dynamics, we define a general dynamics equation parameterized with network embeddings to capture the inherent evolution pattern and impose constraints in a higher structural level on network embeddings. Mutual evolutions of micro- and macro-dynamics in a temporal network alternately affect the process of learning node embeddings. Extensive experiments on three real-world temporal networks demonstrate that M<sup>2</sup>DNE significantly outperforms the state-of-the-arts.

On the other hand, information networks in the real world are often heterogeneous, that is, the network contains multiple types of nodes and relations, called Heterogeneous Information Networks (HIN). Most existing methods usually employ one single model for all relations without distinction, which inevitably restricts the capability of HIN embedding. In this paper, we argue that heterogeneous relations have different structural characteristics, and propose a novel **Relation Structure-aware HIN Embedding** model, called **RHINE**. By exploring four real-world networks with thorough analysis, we present two structure-related measures which consistently distinguish heterogeneous relations into two categories: Affiliation Relations (ARs) and Interaction Relations (IRs). To respect the distinctive structural characteristics of relations, in RHINE, we propose different models specifically tailored to handle ARs and IRs, which can better capture the structures in HINs. Finally, we combine and optimize these models in a unified manner.

Furthermore, the dynamic nature and heterogeneity of networks usually coexist, so-called dynamic heterogeneous information networks. In practice, a real HIN usually evolves over time with the addition (deletion) of multiple types of nodes and edges, and even a tiny change can influence the entire HIN structure and semantic information. In order to capture the

dynamic evolution of the HIN, the conventional HIN embedding methods need to be retrained to get the updated embeddings, which is time-consuming and unrealistic. In this paper, we investigate the problem of dynamic HIN embedding and propose a novel **Dynamic HIN Embedding model (DyHNE)** with meta-path based proximity. As the HIN evolves over time, we naturally capture changes with the perturbation of meta-path augmented adjacency matrices. Thereafter, we learn node embeddings by solving generalized eigenvalue problem effectively and employ eigenvalue perturbation to derive the updated embeddings efficiently without retraining. Experiments on three real-world datasets show that DyHNE outperforms the state-of-the-arts in terms of effectiveness and efficiency.

The powerful modeling capabilities of HIN representation learning have also promoted the development of real application scenarios in the industry. Therefore, this work further studies the friend-enhanced recommendation in the article recommendation scenario of WeChat's "Top Stories", and proposes a **Social Influence Attentive Neural** network, called **SIAN**. In order to fuse rich heterogeneous information, SIAN models recommendation scenarios from the perspective of heterogeneous social networks. In order to fuse rich heterogeneous information, the attentive feature aggregator in SIAN is designed to learn user and item embeddings at both node- and type-levels. More importantly, a social influence coupler is put forward to capture the influence of the friend referral circle in an attentive manner. Experimental results demonstrate that SIAN outperforms several state-of-the-art baselines on two real-world datasets.

**KEY WORDS:** Dynamic Network Heterogeneous Information Network Representation Learning Network Analysis Recommender System

# 目录

第一章 绪论 .....	1
1.1 研究背景及意义 .....	1
1.2 研究现状 .....	3
1.2.1 同质信息网络表示学习 .....	3
1.2.2 异质信息网络表示学习 .....	4
1.2.3 动态网络分析及表示学习 .....	5
1.2.4 网络表示学习的应用 .....	7
1.3 研究内容 .....	7
1.3.1 结合微观和宏观动态性的动态网络表示学习 .....	7
1.3.2 关系结构感知的异质信息网络表示学习 .....	8
1.3.3 动态异质信息网络表示学习 .....	9
1.3.4 用于好友增强推荐的社交影响力专注的神经网络 .....	10
1.4 论文结构 .....	11
1.5 本章小结 .....	11
第二章 相关概念 .....	12
2.1 动态网络 .....	12
2.1.1 动态网络定义 .....	12
2.1.2 动态网络分析 .....	12
2.2 异质信息网络 .....	13
2.2.1 异质信息网络定义 .....	13
2.2.2 异质信息网络模式 .....	14
2.2.3 异质信息网络结构 .....	14
2.3 网络表示学习 .....	15
2.3.1 网络表示学习定义 .....	15
2.3.2 网络表示学习应用 .....	15
2.4 本章小结 .....	16
第三章 结合微观和宏观动态性的动态网络表示学习算法 .....	17
3.1 研究背景 .....	17
3.2 算法设计与实现 .....	19
3.2.1 符号及定义 .....	19

3.2.2 算法框架 .....	20
3.2.3 保持微观动态性的嵌入 .....	20
3.2.4 保持宏观动态性的嵌入 .....	22
3.2.5 模型优化 .....	23
3.3 实验及分析 .....	24
3.3.1 实验设置 .....	24
3.3.2 有效性实验 .....	24
3.3.3 微观动态性实验 .....	26
3.3.4 宏观动态性实验 .....	28
3.3.5 参数分析实验 .....	29
3.4 本章小结 .....	30
第四章 关系结构感知的异质信息网络表示学习算法 .....	31
4.1 研究背景 .....	31
4.2 异质信息网络中关系结构特性分析 .....	33
4.2.1 数据表述 .....	33
4.2.2 隶属关系和交互关系 .....	34
4.3 模型设计与实现 .....	36
4.3.1 符号及定义 .....	36
4.3.2 基本思路 .....	36
4.3.3 建模 ARs 和 IRs 的不同模型.....	37
4.4 实验及分析 .....	38
4.4.1 实验设置 .....	38
4.4.2 节点聚类实验 .....	39
4.4.3 链路预测实验 .....	39
4.4.4 节点分类实验 .....	40
4.4.5 节点推荐实验 .....	41
4.4.6 模型分析实验 .....	42
4.5 本章小结 .....	44
第五章 动态异质信息网络表示学习算法 .....	45
5.1 研究背景 .....	45
5.2 算法设计及实现 .....	47
5.2.1 符号及定义 .....	47
5.2.2 基本思路 .....	48
5.2.3 静态异质信息网络表示学习模型 .....	48

5.2.4 基于谱理论的优化 .....	50
5.2.5 动态异质信息网络表示学习模型 .....	50
5.3 实验及分析 .....	54
5.3.1 实验设置 .....	54
5.3.2 StHNE 模型有效性实验.....	55
5.3.3 DyHNE 模型有效性实验 .....	56
5.3.4 StHNE 模型效率性实验.....	58
5.3.5 DyHNE 模型效率性实验。 .....	58
5.3.6 参数分析实验 .....	60
5.4 本章小结 .....	60
第六章 用于好友增强推荐的社交影响力专注的神经网络 .....	62
6.1 研究背景 .....	62
6.2 模型设计及实现 .....	64
6.2.1 符号及定义 .....	64
6.2.2 模型总览 .....	65
6.2.3 注意力特征聚合器 .....	66
6.2.4 社交影响力耦合器 .....	67
6.2.5 行为预测及模型训练 .....	67
6.3 实验分析 .....	68
6.3.1 实验设置 .....	68
6.3.2 实验结果及分析 .....	69
6.3.3 多方面信息影响分析 .....	70
6.3.4 社交影响力分析 .....	70
6.4 本章小结 .....	72
第七章 总结与展望 .....	73
7.1 总结 .....	73
7.2 展望 .....	74
参考文献 .....	75
致谢 .....	82
攻读学位期间取得的研究成果 .....	83

## 表格索引

表 2-1 部分常见任务描述.....	15
表 3-1 数据集统计信息.....	24
表 3-2 网络重构实验结果.....	25
表 3-3 节点分类实验结果.....	26
表 3-4 时序链路预测实验结果.....	28
表 3-5 网络规模预测实验结果.....	28
表 4-1 数据集统计信息.....	33
表 4-2 节点聚类实验结果.....	39
表 4-3 链路预测实验结果.....	40
表 4-4 节点分类实验结果.....	41
表 5-1 数据集统计信息.....	54
表 5-2 在静态异质信息网络上节点分类实验结果.....	55
表 5-3 在静态异质信息网络上关系预测实验结果.....	56
表 5-4 在动态异质信息网络上的节点分类实验结果.....	57
表 5-5 在动态异质信息网络上的关系预测实验结果.....	58
表 6-1 数据集统计信息.....	68
表 6-2 行为预测实验结果.....	69

## 插图索引

图 1–1 信息网络集合.....	1
图 1–2 网络表示学习的基本范式.....	2
图 2–1 三种常见异质信息网络.....	14
图 2–2 三种常见异质信息网络模式.....	14
图 3–1 时序动态网络中的微观与宏观动态性.....	19
图 3–2 模型 M <sup>2</sup> DNE 总览 .....	20
图 3–3 时序节点推荐实验结果.....	27
图 3–4 趋势预测实验结果.....	29
图 3–5 参数实验结果.....	30
图 4–1 异质信息网络示例以及传统方法和本研究提出方法的比较.....	33
图 4–2 三个数据集中不同关系两端节点的度分布.....	34
图 4–3 节点推荐实验结果.....	42
图 4–4 模型分析实验.....	42
图 4–5 可视化实验.....	43
图 4–6 参数实验.....	44
图 5–1 异质信息网络中基于元路径的一阶和二阶相似性的示例.....	47
图 5–2 StHNE 和 DyHNE 模型结构 .....	48
图 5–3 StHNE 有效性实验结果 .....	58
图 5–4 和对比方法相比 DyHNE 模型效率性 .....	59
图 5–5 和 StHNE 相比 DyHNE 模型效率性 .....	59
图 5–6 模型运行时间.....	59
图 5–7 元路径的融合实验.....	60
图 5–8 表示向量维度实验.....	60
图 6–1 好友增强的推荐场景.....	64
图 6–2 模型 SIAN 总览 .....	65
图 6–3 基于注意力权重的多方面信息影响分析.....	70
图 6–4 基于用户属性的社交影响力分析.....	71



# 第一章 绪论

## 1.1 研究背景及意义

近年来，信息时代快速发展，网络数据呈爆炸式增长。各类信息网络建模了丰富多样的数据，如学术合作网络描述了研究者的合作关系，社交网络刻画了用户的社交行为。通过分析这些信息网络，我们可以充分挖掘数据的隐含信息，从而提升实际应用的表现。根据包含的节点或边的类型是否相同，信息网络可划分为同质信息网络<sup>[1-3]</sup>和异质信息网络<sup>[4-6]</sup>。同质信息网络往往是现实中的网络的一种简化，其包含单一的节点和边类型，而包含不同类型节点和边的异质信息网络可以更加完整自然地对网络数据建模。另一个角度，在现实世界中，网络通常是动态变化的，不断有新的节点和边被增加或删除。因此，信息网络又可以划分为静态信息网络<sup>[7,8]</sup>和动态信息网络<sup>[9-11]</sup>。图 1 - 1 展示了不同类别信息网络的关系集合，本文立足于研究动态同质信息网络、静态异质信息网络以及动态异质信息网络，建模信息网络的本质特性，并在真实应用场景中验证有效性。

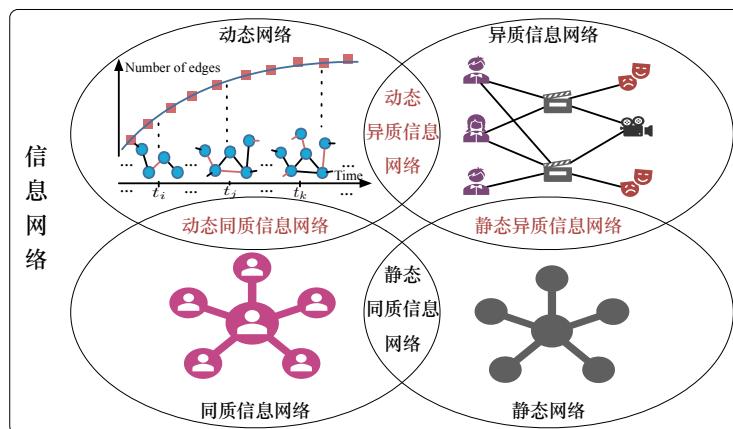


图 1 - 1 信息网络集合

网络数据的分析与研究一直是数据挖掘中的研究热点。通常，我们利用邻接矩阵表示信息网络的结构，并基于矩阵分析方法挖掘网络信息<sup>[11-15]</sup>。然而，现实生活中的信息网络通常是不断增长且稀疏的，传统的网络分析方法很难适用于大规模网络。随着深度学习的广泛应用，一种将网络节点映射到低维空间的方法，即网络的表示学习<sup>[1,16-22]</sup>，为信息网络分析提供了新的研究思路。网络的表示学习旨在将网络中的节点或者边映射到一个低维度的隐含向量空间，从而可以更加灵活地应用于不同的数据挖掘任务中，如节点分类、链路预测和网络可视化等。图 1 - 2 示例了网络表示学习的基本范式，网络表示学习以一个节点数量为  $N$  的网络数据为输入，输出一个大小为  $N * d$  的向量表示矩阵，矩阵每一行表示一个

节点的低维度向量（维度 $d$ 一般较小，如 64 或 128 维）。这些低维度密集向量能够建模网络中复杂的结构和丰富的语义信息，并作为下游机器学习模型（如逻辑回归分类器）的输入，应用于不同的任务场景中。

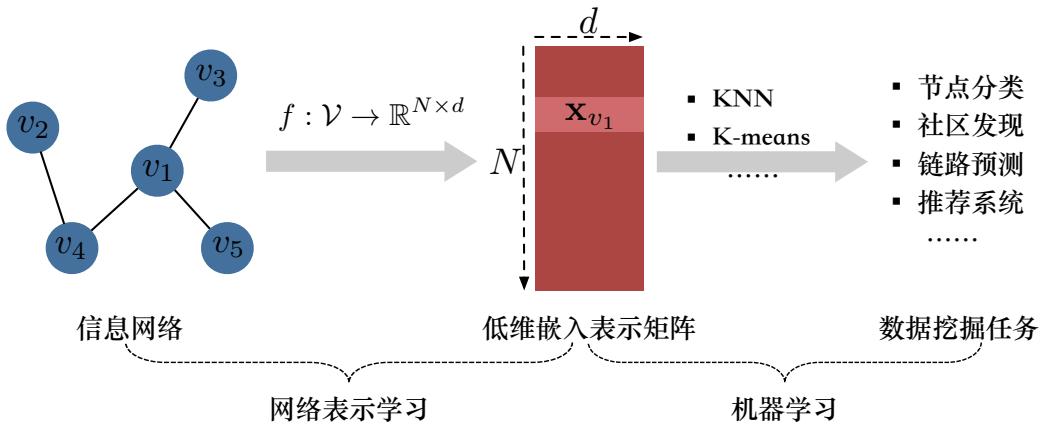


图 1-2 网络表示学习的基本范式

网络表示学习为网络数据分析提供了新颖的研究思路，可以为不同的数据挖掘任务提供包含更丰富信息的特征输入。在同质网络表示学习方面，Perozzi 等人融合随机游走和 skip-gram 模型<sup>[23]</sup>，提出了同质网络表示学习模型 DeepWalk<sup>[24]</sup>。类似的，Grover 等人<sup>[25]</sup>进一步提出了带偏置的随机游走模型，用于学习网络节点的向量表示。而近些年随着深度神经网络的不断发展和成熟，Wang 等人<sup>[26]</sup>基于深度自编码器建模网络的局部和全局结构，从而保留网络的特性。近期备受关注的图神经网络<sup>[17,18,27-31]</sup>基于邻居聚合的思想，建模网络结构和特征信息，有效的编码了网络内部特性，为网络表示学习提供了新思路。虽然大量的同质网络表示学习模型被提出，但他们大多数都假设网络的静止不变的，即节点和边的数量是固定的。然而在实际生活中，网络往往是动态变化的。因此，Cui 等人<sup>[32]</sup>通过将 GSVD<sup>[33]</sup>分解模型扩展为广义特征值问题，实现了在保留网络高阶属性的同时，动态更新网络节点表示。为了更细粒度地捕获动态网络的演变趋势，Zuo 等人<sup>[34]</sup>基于霍克斯过程<sup>[35]</sup>建模了网络的演变模式。遗憾的是，这些方法要么只考虑了静态同质信息网络，要么只建模了网络微观的结构演变。他们忽略了网络在宏观层面的动态变化，即网络规模存在一定的演变模式，而这种演变模式为节点表示提供了更高阶的网络特性信息，这对于动态网络表示学习也是至关重要的。

在异质信息网络表示学习方面，许多研究工作致力于建模多类型节点和边所带来的异质结构和丰富语义。作为一种复杂关系的元结构，元路径<sup>[36]</sup>是一种常用来探索网络结构和语义信息的子结构，其本质是一个连接两类对象的关系组合，例如在学术合作网络中的“作者-论文-作者”元路径表示作者合作关系。基于元路径的随机游走，metapath2vec<sup>[37]</sup>结合 skip-gram 模型学习异质网络的节点表示。HERec<sup>[38]</sup>利用元路径抽取异质网络中的多个同质网络，然后通过融合函数对多个

同质网络的表示进行融合。虽然这些方法探索了异质信息网络的结构和语义信息，但他们存在两个主要问题。首先，现有方法往往忽略了异质信息网络本身的特性，例如复杂多样关系中的结构差异性，不同类型关系的语义区别等。这些特性更能体现异质信息网络的固有特性，因此异质信息网络表示学习需要考虑这些差异化的结构信息，并在其低维向量表示中有所编码保留。其次，现有的异质信息网络表示学习模型大多聚焦于静态网络，即嵌入的网络大小是静止不变的。然而，真实世界的网络往往是动态变化的，而动态异质信息网络的演变又因其异质性更加复杂多样。异质性导致传统的动态同质网络表示学习方法无法有效建模异质信息网络的异质结构和丰富语义，而动态性又要求传统静态异质信息网络表示学习方法能够处理网络的动态演变。因此，如何建模网络中的异质结构和丰富语义，并且捕获网络的动态演变，是动态异质信息网络表示学习一个亟待解决的问题。

异质信息网络表示学习的优越性不仅在学术界得以体现，工业界对其应用前景同样非常期待。在工业界中，异质信息网络能够有效地建模众多应用场景，例如 Facebook 和微信的社交网络、Amazon 和阿里巴巴的电商网络等等。例如，阿里巴巴团队构建了图神经网络平台 AliGraph<sup>[39]</sup>，其结合异质信息网络表示学习模型，用于复杂多样的电商推荐场景。针对微信“看一看”的推荐场景，腾讯微信团队改进了传统 look-like 算法<sup>[40]</sup>，对二部异质图进行用户表示学习，进而提升推荐效果。整体上，这些方法利用异质信息网络表示学习，建模了特定业务需求，为真实场景带来了显著提升。因此，如何充分结合异质信息网络表示学习与真实业务场景，在真实场景中验证异质信息网络表示学习的有效性，是一个从科学到业务落地过程中需要考虑的重要问题。

总体来说，网络表示学习的快速发展，为下游数据挖掘任务提供了丰富的特征，从而提升了实际应用场景的表现结果，使得其成为了学术界和工业界的研究热点。然而，现实世界中的信息网络往往是动态的、异质的，这对传统方法提出了巨大挑战。因此，本论文旨在构建新颖的网络表示学习算法，其有能力建模同质网络的动态演变趋势，捕获异质网络的异质结构和丰富语义，处理异质网络的动态节点表示，以及在实际应用场景中验证异质网络表示学习的潜在能力。

## 1.2 研究现状

### 1.2.1 同质信息网络表示学习

同质信息网络包含单一的节点和边类型，被广泛应用于刻画物体间的联系，例如描述人与人之间关系的社交网络。近年来，同质信息网络表示学习逐渐成为

分析研究同质信息网络的重要方法。粗略地，现有同质信息网络表示学习方法可以分为三类：基于矩阵分解、基于随机游走，以及基于深度神经网络的模型。

基于矩阵分解的网络表示学习算法，以矩阵的形式表示网络中节点之间的连接，并通过矩阵分解的方法获得节点的向量表示。例如，谱聚类算法<sup>[41-43]</sup>利用邻接矩阵或拉普拉斯矩阵的前  $k$  个特征向量或奇异向量来表示节点的  $k$  维向量表示。优于传统矩阵分解方法往往存在时间和空间复杂度较大的缺点，从而无法实现大规模网络的应用。近年来，一些方法尝试降低矩阵分解的复杂度，例如 GraRep<sup>[44]</sup> 考虑一种特殊的关系矩阵，利用 SVD 分解得到节点的低维向量表示。为了保留同质网络中的高阶相似性，基于矩阵分解的 HOPE 方法<sup>[45]</sup>构建不同相似性矩阵（如卡茨指数、相同邻居等），并对其进行 SVD 分解，从而获得节点的低维表示。

基于随机游走的方法，通过在网络上进行随机游走，以获得网络结构信息，从而将网络中节点嵌入到低维空间。典型地，两个基于随机游走的同质网络表示学习方法：DeepWalk 和 node2vec。DeepWalk 算法<sup>[24]</sup>利用网络结构中的随机游走序列，结合 skip-gram 模型将节点视为单词，游走序列视为句子，学习节点的向量表示。类似地，node2vec<sup>[25]</sup>通过带偏置的随机游走，获得节点序列，并最大化序列中节点的共现概率保留网络的高阶相似性。基于随机游走的模型降低了传统矩阵分解算法的复杂度，使得其可以适用于分布式和大规模网络。

随着深度学习的发展，基于深度神经网络的模型在同质网络表示学习方面也取得了较好的表现。例如，基于深度自编码器，SDNE<sup>[26]</sup>对数据中的非线性结构进行建模。最近备受关注的图神经网络<sup>[29-31]</sup>基于卷积神经网络的结构，聚合图结构中的邻居信息，有效建模了结构和特征信息，为网络表示学习提供了新思路。GCN<sup>[27]</sup>、GraphSAGE<sup>[17]</sup>和 GAT<sup>[28]</sup>都是将图像数据上的卷积操作泛化到图结构数据上，其关键是学习一个函数，通过聚合自身特征及邻居特征，来生成节点的新向量表示。区别在于，GCN 从谱域的角度，通过图拉普拉斯矩阵聚合邻居信息，GraphSAGE 通过采样固定数量的邻居，实现高效邻居聚合，而 GAT 通过 attention 机制区分聚合邻居的特征信息，从而进一步提升了节点的表示能力。

此外，还有一些其他同质网络表示学习方法致力于保留网络的特殊属性或结合额外信息，例如 LINE<sup>[46]</sup>建模网络结构的一阶相似性和二阶相似性，TADW<sup>[47]</sup> 基于矩阵分解将节点的文本特征信息引入了网络表示学习中，而 CANE 模型<sup>[48]</sup>结合节点文本信息，将节点的表示向量构建为文本表示向量和结构向量的组成。

### 1.2.2 异质信息网络表示学习

不同于同质信息网络仅包含单一的节点和边类型，异质信息网络中的节点类型和边类型是多样的，其更加完整自然地建模了网络数据中的内在联系和结构。

类似于同质信息网络表示学习方法，当前的异质图表示学习同样可以粗略地分为三种类型：基于分解、基于随机游走、以及基于深度神经网络的方法。

异质信息网络表示学习的最直接做法是忽略节点和边类型，直接应用同质信息网络表示学习模型，但这必然会导致信息的损失，降低下游任务的表现。因此，基于分解的思路，许多研究者将异质信息网络拆分为多个同质信息网络，进而对多个同质网络进行嵌入学习。HERec<sup>[38]</sup>基于异质信息网络中的元路径抽取网络中的多个同质网络，然后融合多个同质网络表示向量。EOE<sup>[49]</sup>将二分图网络分解为两个同质信息网络，从而对各子图内节点对和子图间节点对同时进行表示学习。PTE<sup>[50]</sup>以半监督的方式学习文本的嵌入表示，其将从文本中构建的异质网络分解成三个子网络。上述方法均尝试将复杂的异质信息网络分解为多个简单图，从而较好地克服了异质性带来的挑战。

类似于同质网络表示学习，基于随机游走的方法在图上进行随机游走获得游走序列，进而学习节点的嵌入表示。但是，由于异质信息网络中的多类型节点和边关系，通常采用基于元路径的随机游走策略。具有代表性的，基于元路径的随机游走模型有 metapath2vec<sup>[37]</sup>、HINE<sup>[51]</sup>、HIN2vec<sup>[52]</sup>和 ESim<sup>[53]</sup>等。metapath2vec 根据给定的元路径进行随机游走，进而利用 skip-gram 算法将网络结构信息编码到节点的向量表示中。HINE 基于元路径随机游走来计算节点间的相似性，并将其作为监督信息来指导节点的向量表示。考虑到节点类型的区别以及节点间的复杂关系，HIN2vec 通过刻画节点对及其之间的边关系来学习节点及元路径的向量表示。针对异质信息网络中节点相似性度量问题，ESim 在异质信息网络上基于元路径进行随机游走，并通过最大化元路径实例的概率学习节点的向量表示。

相对于浅层模型，深度模型对非线性关系建模的优越性，使其能够抽取更为复杂的语义信息。基于深层自动编码器，SHINE<sup>[54]</sup>建模社交网络、情感网络以及画像网络中的异质信息，进而对这些信息联合编码、融合得到节点的低维表示向量。BL-MNE<sup>[55]</sup>采用自动编码机编码不同元路径下的异质结构信息，并通过转移矩阵融合不同的信息。为了建模文本和图像并存的异质信息网络，HNE<sup>[56]</sup>利用卷积神经网络和多层次感知器抽取文本和图像数据的隐含特征，并通过转移映射矩阵将不同类型的信息投射到同一空间，进而用低维向量表示内在信息。近期的图卷积神经网络同样也适用于异质网络表示学习，例如 Wang 等人<sup>[57]</sup>提出基于节点级别和语义级别的异质图注意力网络，来捕捉异质网络中的复杂结构和丰富语义。

### 1.2.3 动态网络分析及表示学习

在现实生活中，信息网络往往是动态变化而非一成不变的，即图中节点和边随时间增加或减少。例如社交网络不断有新的用户加入或退出，电商网络不断有

新的商品上线或下线等。针对动态网络的研究已经吸引了广泛的关注，其相关工作可大致分为两大类别：动态信息网络的演变分析和动态信息网络的表示学习。

在动态网络的演变分析方面，相关研究旨在建模动态信息网络的演变趋势，实现图的演变趋势的准确预测<sup>[58-60]</sup>。具体地，Bass 模型<sup>[61]</sup>和 SI 模型<sup>[62]</sup>发现了动态图在早期演变过程中的 sigmoid 型增长趋势，建模了动态信息网络中节点的指数型的演变模式。基于对真实社交网络的动态性分析，Zang 等人提出了 NetTide 模型建模动态社交图的演变模式，其既能捕获节点的动态演变也可以建模图中边的增长模式<sup>[63]</sup>。已有的大部分工作聚焦于动态信息网络的增长，然而图中节点和边的衰减也是同样重要的。Zhang 等人<sup>[64]</sup>研究了真实动态社交图中群组的增长和退出模式，其发现了扩散与非扩散并存于群组加入过程；而退出群组呈幂律型衰减，并提出了一个动态模型 ComeNGo 建模解释了动态信息网络的演变模式。

在动态信息网络表示学习方面，相关研究目的在于将动态信息网络中的节点映射到一个低维度的表征向量空间，该空间既能保留动态信息网络本身的拓扑结构属性，同时能够捕获网络的动态演变趋势<sup>[18,20]</sup>。该方面工作又可大致分为两类：基于快照网络的网络嵌入和建模网络时序演变的网络嵌入。具体地，基于快照网络的网络嵌入的基本思想是利用网络快照描述动态信息网络的演变过程，并为每一个网络快照学习其对应的隐含特征表示。针对大规模动态信息网络的演变，重新训练模型以获得最新的节点表示是耗时且不实际的，DANE<sup>[65]</sup>和 DHPE<sup>[32]</sup>基于谱聚类和矩阵摄动理论，实现了动态信息网络嵌入表示的高效且有效的更新。Song 等人<sup>[66]</sup>扩展基于 skip-gram 的网络嵌入模型，提出了一个通用的动态信息网络嵌入框架。通过建模动态信息网络中的三角闭合过程，DynamicTriad<sup>[67]</sup>捕获了动态信息网络演变过程中的拓扑结构信息和演变模式。由于快照网络只能反应图在某一时间间隔内的网络结构，其以粗粒度的方式建模了动态图的时序演变信息。因此，第二类工作，即建模动态信息网络的时序演变，在一定程度上缓解了这一问题。动态信息网络的时序演变以更细粒度的方式描述图的演变模式，能够同时嵌入图的时序信息和拓扑结构。具体地，考虑到动态信息网络的结构形成过程，HTNE<sup>[34]</sup>通过邻居形成序列描述动态信息网络的拓扑结构的变化，并基于霍克斯过程捕获动态信息网络的时序演变，学习图中节点的隐含特征表示。更进一步地，Trivedi<sup>[68]</sup>等人将动态信息网络中的时序过程划分为关联过程和交流过程，并基于时序点过程提出了一个深度表示学习框架。

总体来看，现有的动态信息网络研究的相关工作大多只考虑了网络中的部分演化机制，例如只建模网络的增长模式或只考虑网络中节点的演变模式。然而真实的网络动态性还体现在宏观层面的演变，比如网络规模的演变等。如何在动态信息网络嵌入中考虑宏观演变特性，是动态信息网络表示学习中的一个重要问题。

此外，上述工作大多忽略了网络的异质性，这必然会丢失动态信息网络的内在特性，导致网络嵌入表示的表现欠佳。因此，如何更充分地研究建模动态信息网络的演化机制，全面考虑网络的异质性，将是一个重要且具有实际意义的研究方向。

### 1.2.4 网络表示学习的应用

得益于对异质结构和丰富语义的强大建模能力，异质信息网络表示学习很好地刻画了网络中不同类型节点之间的复杂关系，并在各类工业应用场景落地 上都取得了巨大突破。其中，异质信息网络表示学习在推荐系统中的应用最为广泛，也是本论文的实际场景应用，因此以下对该方面进行系统总结。

在推荐系统方面，研究者致力于利用异质信息网络建模复杂的推荐场景，进而提升推荐效果。例如，基于异质信息网络中不同的元图结构，Zhao 等人<sup>[69]</sup>构建多个交换矩阵，进而利用矩阵分解为用户和商品分别学习多个不同的隐向量，最后经过因子分解机预测用户对商品的评分，并产生推荐结果。基于深度模型的 JRL 模型<sup>[70]</sup>利用异质的信息（如评分、评论、图片等）学习用户和商品的表示向量，为用户推荐 top-N 的物品。Han 等人<sup>[71]</sup>提出了深度神经网络的协同过滤模型 NeuACF 来学习方面级潜在因子，并进行精准推荐。基于元路径的上下文信息，MCRec<sup>[72]</sup>设计了 co-attention 机制，并通过优先级采样选择高质量元路径实例，从而有效地学习到用户、商品和基于元路径的上下文表示。

## 1.3 研究内容

### 1.3.1 结合微观和宏观动态性的动态网络表示学习

在时序动态网络中，节点之间边的建立是按时间顺序排列的，并且网络规模也是随着时间，以某种显著的分布不断增长的。举例来说，在不同年份，研究者和他人合作，这会导致序列化的合作事件和不断增长的网络规模。因此，一个时序网络自然地表示出了网络的演变，不仅包括细粒度的网络拓扑结构的演变，还包括宏观的网络规模趋势的变化。用隐空间嵌入时序网络，考虑网络微观拓扑结构和宏观规模变换，对于实际应用是非常重要的。

在微观层面，时序动态网络的结构由网络中边的建立而驱动，其实际上是一个按时间顺序排列的事件序列，且每个事件由两个节点参与。微观动态的演变过程，细致地描述了在不同时刻节点间边的形成过程，并且解释了“为何网络在  $t$  时刻会演变为如此结构”。建模这种微观动态性，使得学到的节点表示更准确地捕获时序网络的演变，这对于下游的时序网络任务是很有益的。

在宏观层面，时序动态网络另一个显著的特性是网络规模通常以明显的分布随时间演变，例如 S-形曲线或幂律形模式。因此，网络规模，即边的数量，随时间增长且服从一定的内在准则，而非随机生成的。这样的宏观动态性反映了时序网络的内在演变模式，并在更高的结构层次对网络嵌入施以约束。也就是说，宏观动态性决定了：随时推进，由微观动态性嵌入而新形成的边的数量。宏观动态性的融入，提供了有价值且有效的演变信息，从而可以提升网络嵌入保持网络结构和演变模式的能力，也会大大增强网络嵌入的泛化能力。因此，学习到的表示空间能否编码时序网络中的宏观动态性应该是时序网络嵌入的一个重要要求。

针对动态网络的微观和宏观演变，本论文提出了一种新颖的融合微观动态性和宏观动态性的动态网络表示学习模型 M<sup>2</sup>DNE。具体地，为了建模时序网络中边的建立过程，即按时间顺序排列的事件（也就是微观动态性），M<sup>2</sup>DNE 模型通过用网络表示表示参数化条件强度函数，设计了一个时序注意力点过程，其利用层级时序注意力捕获细粒度的结构和时序特性。为了建模时序网络规模的演变模式（即宏观动态性），M<sup>2</sup>DNE 模型定义一个通用动力学方程为网络表示表示的非线性函数，其在更高的结构层次对网络嵌入施以约束，并且很好地耦合了时序网络中的动力学分析和表示学习。最后，通过整合微观和宏观演变保留网络的动态性，并且联合优化两者。由于微观和宏观动态性交互地演变且迭代影响节点表示的学习，M<sup>2</sup>DNE 能够统一地捕获拓扑结构的形成过程和网络规模的演变模式。

### 1.3.2 关系结构感知的异质信息网络表示学习

异质信息网络包含多种类型的节点和边，以学术网络为例，网络中通常包括 Author, Paper, Conference 以及 Term 四种类型的节点，Author-Paper, Paper-Conference 以及 Paper-Term 三种类型的边。此外，异质信息网络的重要特性之一就是：网络中包含丰富且多样的语义信息，这些语义信息可以形象地刻画网络的实际意义，捕获这些语义信息有利于各类数据挖掘任务。在异质信息网络中，通常使用元路径(meta-path)描述网络中的复杂关系和语义信息。基于不同元路径，节点具有不同的路径语义的连接关系。多种类型的节点和边导致异质信息网络的拓扑结构更加复杂，并有着明显不同的结构特征，需要以不同的模型区分对待。

对于异质网络嵌入表示，考虑关系的不同的结构特征是不容易的。主要有以下几个挑战：(1) 如何区分异质网络中关系的结构特征？在异质网络中，多样的关系（原子关系和元路径）包含不同的结构。(2) 如何捕获不同类别关系的不同的结构特征？由于多样的关系有不同的结构，以一个模型建模它们可能会导致信息的丢失。因此需要特别设计合适的模型，用于捕获有差异的结构特征。(3) 建模不同关系的不同模型应该能够容易地融合，并且可以联合优化求解。

为了解决上述挑战，本论文提出了一个新颖的异质信息网络嵌入模型，即关系结构感知的异质信息网络嵌入（RHINE）。具体地，经过透彻的数学分析，RHINE 探索了异质信息网络中关系的结构特征，并提出了两个结构相关的度量方法。这两个方法可以将多样关系一致地划分为两类：表示 one-centered-by-another 结构的隶属关系（Affiliation Relations, ARs）和表示 peer-to-peer 结构的交互关系（Interaction Relations, IRs）。为了捕获关系中差异的结构特征，提出了两种特别设计的模型。对于 ARs 关系，关系中的节点共享一些相似的特性，RHINE 模型计算节点间的欧式距离，以此保证节点在隐含空间中直接相近。另一方面，对于桥接两个对等节点的 IRs 关系，RHINE 模型将这类关系建模为节点间的翻译。由于两个模型在数学形式上是一致的，它们可以以统一的方式联合优化求解。

### 1.3.3 动态异质信息网络表示学习

在动态异质信息网络表示学习中，有两个根本问题需要被仔细考虑。（1）第一个问题时，如何有效地在动态环境中保留网络的拓扑结构和语义关系。由于拓扑结构和语义关系是异质信息网络中两个最重要且直接的信息，它们从本质上保证了节点表示的有效性。当一个新的节点加入到异质信息网络中，以节点为中心的局部结构已发生变化。同时，这种变化会通过不同的元路径逐渐传递到所有的节点，这会导致全局结构的变化。此外，新节点不仅会和邻居建立直接连接，同时也会通过各种元路径和其他节点建立复杂的关系，这必然会影响异质信息网络中复杂的语义关系。因此，随着动态异质信息网络的演变，网络的拓扑结构和语义关系都会发生改变。对于一个有效的动态异质信息网络嵌入模型，同时建模这种变化并编码结构和语义信息是很有必要。（2）第二个问题，当网络发生演变时，如何高效地更新节点的表示而不重新训练更新异质信息网络。对于每个时间步，重新训练一个异质信息网络嵌入模型是获得最优节点表示的最直接方法。然而，显然地，这种策略是非常耗时的，尤其当网络的结构发生很微小变化的时候。在大数据时代，重新训练的方式是不实际的。因此，构建一个切实可行的动态异质信息网络嵌入方法，以此更新节点的表示是非常有必要的。

因此，本论文提出了一个动态异质信息网络嵌入的模型（DyHNE）来有效且高效地学习节点的表示。受广泛用于捕获系统变化的扰动理论的启发，DyHNE 模型可以通过求解广义特征值问题来学习节点嵌入，并利用特征值扰动对异质信息网络的演化进行建模。沿着这个研究路线，首先采用元路径增强的邻接矩阵来建模异质信息网络的拓扑结构，然后构建一个基础静态异质信息网络嵌入模型，即 StHNE 模型，来保留基于元路径的一阶相似性和二阶相似性。因此，StHNE 模型可以更好地捕获动态异质信息网络中的结构和语义信息。为了捕获异质信息

网络的演变，模型利用多个元路径增强邻接矩阵的扰动以自然的方式建模异质信息网络的结构和语义的变化。更进一步地，基于特征值摄动理论，DyHNE 模型捕获了网络的变化并高效地求解网络节点的表示。

### 1.3.4 用于好友增强推荐的社交影响力专注的神经网络

学习异质网络的嵌入表示能够较好建模网络中不同类型节点之间的复杂关联，应用于各类业务场景。因此，异质信息网络表示学习也越来越受到工业界的青睐。本论文结合异质信息网络表示学习和推荐系统，微信“看一看”的文章推荐场景下，研究好友增强的推荐问题，以验证异质信息网络的表达能力。

如今，随着在线社交网络的兴起，人们更愿意在社交平台上积极表达自己的观点并与朋友共享信息，同样也会看到朋友愿意分享的显式的交互记录，因此朋友成为重要的信息来源和高质量的信息过滤器。朋友互动过的项目（共享的，喜欢的等）会对用户产生重大影响，这很可能会成为用户未来的兴趣所在。在工业界，许多推荐系统专注于好友的社交影响（例如，YouTube 中的信息推荐流和微信中的看一看）。受启于社会化推荐，本论文结合微信看一看的真实业务场景，形式化了一种好友提升的推荐场景（Friend Enhanced Recommendation，FER）。该场景与传统的社交推荐有两个主要区别：（1）该推荐场景仅向用户推荐他/她的朋友与之互动的内容，将朋友视为提供更多高质量项目的高质量信息过滤器。

（2）与项目互动的所有朋友都明确显示给推荐项目的用户，这突出了显式社会因素的至关重要性，并提高了用户行为的可解释性。

新颖的推荐场景对异质信息网络表示学习提出了一些挑战：（1）如何从多方面的异质因素中提取关键信息？好友提升的推荐场景涉及多个异质影响因素，例如物品内容（新闻内容），好友信息及其好友与新闻的交互交互。这些影响因素决定了用户的行为，因此在该场景下的异质信息网络表示学习不仅需要学习用户对商品的偏好，还需要捕获不同影响因素对用户的影响程度，从而预测用户对物品（新闻）的行为。（2）如何利用显示的好友推荐信息？好友提升的推荐场景极大地强调了社交信息在推荐中的重要性，但很少有工作在实际推荐中探索显式社交信息的影响，因此需要设计模型来充分利用该场景中的显式的朋友推荐信息。

为了解决这些问题，本论文提出了一种新颖的社交影响力专注的神经网络（SIAN）。具体来说，好友提升的推荐场景被建模为异质信息网络上的用户和物品的交互预测任务。为了捕获异构信息和多方面的影响，SIAN 模型设计了一个由节点级和类型级聚合组成的细心特征聚合器。作为所提出模型的关键组成部分，影响力耦合器通过显式社会关系对耦合影响力的扩散进行有效模。通过学习到的用户，商品和社交影响的潜在表示，SIAN 可以预测用户与商品交互的概率。

## 1.4 论文结构

本论文的主要内容，由以下几个部分组成：

第一章是绪论。首先主要介绍本论文的研究背景及意义；紧接着，系统总结与本论文相关的国内外研究现状及相关工作；最后，该章节简略阐述本论文的四个研究内容及论文的整体组织结构。

第二章是相关技术。这一章主要介绍了动态网络、异质信息网络以及网络表示学习的相关概念和技术，并对本论文相关的技术基础进行了系统介绍。

第三章介绍第一个研究点，即结合微观和宏观演变的动态网络表示学习算法。本章首先介绍模型的研究背景及核心思想，其后是对模型算法的详细叙述，最后展示该模型在多个数据集上的实验结果及分析。

第四章介绍第二个研究点，即关系结构感知的异质信息网络表示学习算法。首先，对算法的研究背景及创新贡献进行概述。其次，介绍算法框架及模型细节。最后是对模型有效性的实验验证，包括多个数据集、多个任务的结果展示与分析。

第五章介绍第三个研究点，即动态异质信息网络的增量表示学习算法。首先概述算法的研究背景及基本思路，紧接着是对算法核心设计的详细叙述，最后介绍在多个数据集上模型的实验表现，并对实验结果进行分析。

第六章介绍第四个研究点，即针对微信看一看推荐场景的，社交影响专注的神经网络模型。本章首先介绍模型的业务场景及核心思路，其后是对模型架构的详细描述，最后结合在真实数据上的实验结果，进行实验分析。

第七章总结本论文的研究工作，并展望下一步研究方向和目标。

## 1.5 本章小结

本章内容为绪论。在 1.1 节，介绍了该课题的研究背景和意义；1.2 节总结了该课题相关的国内外研究现状。1.2.1 小节概述了同质信息网络表示学习算法，1.2.2 小节总结了异质信息网络表示学习的相关工作，1.2.3 小节对动态网络分析及表示学习的最新进展进行了总结讨论，1.2.4 小节调研了网络表示学习在工业界的实际应用。1.3 节概述了本文的四个研究点，包括：结合微观和宏观演变的动态网络表示学习算法（1.3.1 小节），关系结构感知的异质信息网络表示学习算法（1.3.2 小节），动态异质信息网络的增量表示学习算法（1.3.3 小节），以及针对微信看一看推荐场景的，社交影响专注的神经网络模型（1.3.4 小节）。

## 第二章 相关概念

### 2.1 动态网络

#### 2.1.1 动态网络定义

在真实世界中，网络往往是动态变化的，即节点和边随时间而增加或减少。例如社交网络不断有新的用户加入或推出。根据网络动态变化的刻画程度，可以从两个角度定义动态网络：快照动态网络和时序动态网络<sup>[67,68]</sup>。

(1) 快照动态网络描述了网络离散的演变模式，将网络演变过程分解为不同时间片的网络快照。形式化地，

$$\mathcal{G} = \{\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_{\mathcal{T}}\} \quad (2-1)$$

这里 $\mathcal{G}$ 表示从时刻1到时刻 $\mathcal{T}$ 的一系列网络快照组成的动态网络，其中 $\mathcal{G}_t = \{\mathcal{V}_t, \mathcal{E}_t\}$ 表示在 $t$ 时刻的网络快照， $\mathcal{V}_t$ 和 $\mathcal{E}_t$ 分别表示在 $t$ 时刻网络的节点集合和边关系集合。快照动态网络从较为粗糙的粒度对网络的演变模式进行了刻画，更着重于网络结构上的前后结构变化，而忽略某一时刻内的细粒度演变。

(2) 时序动态网络刻画了网络连续的演变模式，对网络中每条边的形成存在一个对应的时间戳，其从更细粒度建模了网络的演变趋势。形式化地，

$$\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{T}) \quad (2-2)$$

其中 $\mathcal{V}$ 和 $\mathcal{E}$ 分别表示节点和边的集合， $\mathcal{T}$ 是时间戳序列。因此，时序动态网络表示一个带有时间戳的边序列，其中每条时序边 $(i, j, t) \in \mathcal{E}$ 在某一特定时间 $t$ 连接节点 $i$ 和节点 $j$ 。需要注意的是，在不同时刻节点 $i$ 和节点 $j$ 可能建立多条边，此时 $(i, j, t)$ 是一条不同的时序边。

可以看到，快照动态网络将网络演变过程描述为某一时间段内的网络演变的积累，即对应当前时刻的快照网络，其粗粒度地刻画了网络的演变趋势。不同的，时序动态网络从网络中每条边的建立时刻建模网络的演变，从更细粒度的角度考虑了动态网络中边的形成过程。两者各有利弊，对应不同的应用场景。

#### 2.1.2 动态网络分析

信息网络的动态性刻画了真实世界中关系数据的变化，许多研究者致力于构建动力学模型，分析动态信息网络的演变趋势，实现对网络演变趋势的准确预测<sup>[62,63]</sup>。较早的，传染病模型<sup>[62]</sup>描述了传染病的传播过程，可以看作是对动态网络分析研究的经典动力学模型。最基本的传染病动力学模型包括SI模型、SIS模

型，以及 SIR 模型。以 SI 模型为例，其假设总人数  $N$  是恒定不变的，从  $t$  时刻经过  $\Delta t$  时间间隔后，网络中健康人数逐步被感染，其形式化表示为：

$$N[i(t + \Delta t) - i(t)] = [\lambda s(t)] N i(t) \Delta t \quad (2-3)$$

其中病人和健康人的比例分别为  $i(t)$  和  $s(t)$ 。每个病人每天有效接触人数为  $\lambda$ ，且使接触的健康人致病。

SI 模型描述了传染病的传播过程，可以看作是对动态网络增长趋势的建模。近年来，有研究者提出了更为复杂的动力学模型，旨在分析信息网络的动态性。例如，Zang 等人<sup>[63]</sup>研究了数个真实网络的增长情况，观察到网络中节点和边关系的幂律增长趋势。对于一个在  $t$  时刻拥有  $n(t)$  用户（节点）的社交网络，在网络形成早期有较多的用户（节点）加入，而在后期网络逐渐致密，其增长趋势也逐渐变缓。形式化的，其演变增长模式可以定义为如下：

$$\frac{dn(t)}{dt} = \frac{\beta}{t^\theta} n(t)(N - n(t)) \quad (2-4)$$

其中  $N - n(t)$  定义了潜在待加入网络的用户（节点）， $\frac{\beta}{t^\theta}$  表示随时间衰减的网络吸引力。该模型刻画了动态网络的幂律级增长模式，建模了网络的宏观演变趋势。

## 2.2 异质信息网络

### 2.2.1 异质信息网络定义

区别于同质信息网络，异质信息网络包含多类型的节点和边关系，其融合了复杂的结构信息和丰富的语义信息。因此，异质信息网络可以定义为如下：

#### 定义 2.1 异质信息网络

一个异质信息网络通常被定义为一个图  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{O}, \mathcal{R})$ ，其中  $\mathcal{V}$  和  $\mathcal{E}$  分别表示节点和边关系的集合。每个节点  $v \in \mathcal{V}$  和边  $e \in \mathcal{E}$  分别关联一个类型映射函数  $\phi: \mathcal{V} \rightarrow \mathcal{O}$  和  $\varphi: \mathcal{E} \rightarrow \mathcal{R}$ ，其中  $\mathcal{O}$  和  $\mathcal{R}$  表示节点和边的类型集合。当  $|\mathcal{O}| + |\mathcal{R}| > 2$  时，图  $\mathcal{G}$  称为异质信息网络，否则，即为同质信息网络。

图 2-1 示例了常见的三种异质信息网络，包括学术合作网络、电影推荐网络、以及商店点评网络。在学术合作网络中（图 2-1(a)），包括四种节点类型，即作者（Author）、论文（Paper）、会议（Conference）和关键词（Term）。四类节点之间存在多种类型的关系，例如作者-论文描述写作关系，论文-会议描述论文发表关系等；在电影推荐网络中（图 2-1(b)），包括用户（User）、电影（Movie）、导演（Director）和演员（Actor）四类节点，以及多种类型的连接关系；在商店点评网络中（图 2-1(c)），包括用户（User）、商店（Business）、评论（Review）等多种类型的节点，以及多种类型的边关系。

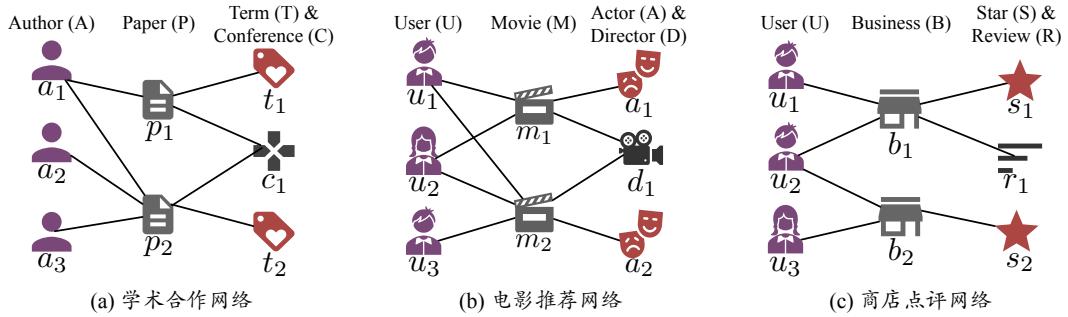


图 2-1 三种常见异质信息网络

### 2.2.2 异质信息网络模式

异质信息网络的多类型节点和边为网络分析提供了丰富的结构和语义信息，但同时也使得异质信息网络的分析更有挑战。许多研究者提出利用网络模式来形式化描述网络的结构及语义，以便更好地了解网络中的对象类型和边类型。

#### 定义 2.2 网络模式

网络模式描述异质信息网络的元结构，是异质信息网络  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{O}, \mathcal{R})$  的元模版。形式化地，网络模式定义为  $\mathcal{S} = (\mathcal{O}, \mathcal{R})$ ，其中  $\mathcal{O}$  和  $\mathcal{R}$  是节点和边的类型集合，节点和边分别关联各自的映射函数  $\phi: \mathcal{V} \rightarrow \mathcal{O}$  和  $\varphi: \mathcal{E} \rightarrow \mathcal{R}$ 。

图 2-1 中的异质信息网络对应的网络模式如图 2-2 所示。以图 2-2(a) 为例，网络模式连接对象类型，对象之间存在不同类型的边表示不同的语义关系。

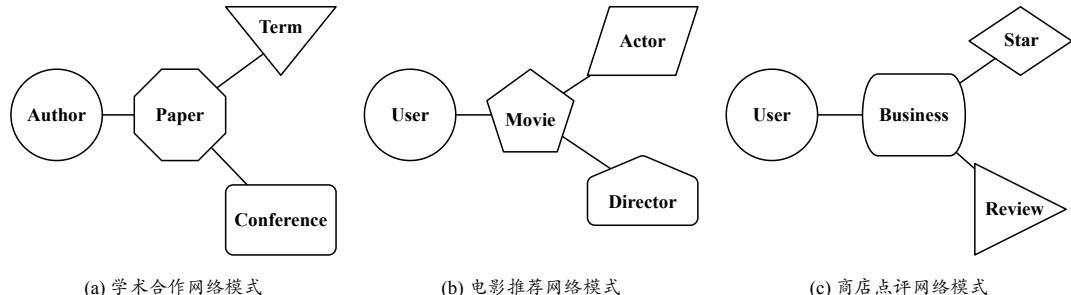


图 2-2 三种常见异质信息网络模式

### 2.2.3 异质信息网络结构

在异质信息网络中，两个对象可以通过不同的语义路径（称为元路径）连接，这些语义路径描述了对象之间的复合关系。形式化地，元路径可以定义如下。

#### 定义 2.3 元路径

对于异质信息网络  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{O}, \mathcal{R})$ ，长度为  $l$  的元路径可以定义为一种复合关系： $\mathcal{P} = o_1 \xrightarrow{r_1} o_2 \xrightarrow{r_2} \dots \xrightarrow{r_l} o_{l+1}$ ，其中  $o_i \in \mathcal{O}$  且  $r_i \in \mathcal{R}$ 。如果两个对象/节点之间有且只有一种关系类型，上述元路径可以简写为  $\mathcal{P} = o_1 o_2 \dots o_{l+1}$ 。

举例来说，在学术合作网络（如图 2-1(a) 所示）中，存在多条元路径。元路径 APA 描述了作者合作关系，APCPA 描述了参加同一会议的关系。而  $a_1 p_1 a_2$  表示元路径 APA 的一个具体路径实例，表示作者  $a_1$  和作者  $a_2$  合作了一篇论文  $p_1$ 。

## 2.3 网络表示学习

### 2.3.1 网络表示学习定义

网络表示学习旨在将网络中的每个节点表示为一个低维的隐含向量，该向量能够编码节点在网络中的结构信息、属性信息等。以节点的低维向量表示为输入，下游模型可以进行多种数据挖掘任务。如图 1-2 所示，网络表示学习连接了原始网络数据和下游数据挖掘任务，为下游任务提供优质输入，以便提升表现效果。

形式化地，给定网络  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{O}, \mathcal{R})$ ，其中  $\mathcal{V}$  和  $\mathcal{E}$  分别表示节点和边关系的集合， $\mathcal{O}$  和  $\mathcal{R}$  是节点和边的类型集合。当  $|\mathcal{O}| + |\mathcal{R}| > 2$  时， $\mathcal{G}$  称为异质信息网络，否则，即为同质信息网络。对于网络  $\mathcal{G}$  中的每个节点  $v \in \mathcal{V}$ ，网络表示学习目的在于学习一个映射函数  $f: v \rightarrow \mathbf{x}_v \in \mathbb{R}^d$ ，其中节点表示向量维度  $d << |\mathcal{V}|$ ，映射函数  $f$  将每个节点  $v \in \mathcal{V}$  映射到一个低维的向量空间并保留网络结构、语义等信息。

### 2.3.2 网络表示学习应用

网络表示学习输出每个节点的低维向量表示，其可以作为各类数据挖掘任务的输入特征。常见的网络表示学习应用有节点聚类、节点分类和链路（关系）预测、推荐系统等，部分任务描述如表 2-1 部分常见任务描述所示。

表 2-1 部分常见任务描述

任务	描述	方法	评价指标
聚类	节点向量表示作为特征，输入聚类器	K-means、谱聚类、层次聚类等	NMI、困惑度等
分类	节点向量表示作为特征，输入分类器	如逻辑回归、朴素贝叶斯等。	F1、Accuracy 等
链路预测	预测节点间是否存在边连接或某关系	计算两个节点表示向量的相似性	AUC、F1 等

## 2.4 本章小结

本章节主要介绍了本文研究内容的相关概念及技术。2.1 节概述了动态网络的相关内容，包括 2.1.1 小节对动态网络的形式化定义，和 2.1.2 小节对动态网络分析技术的介绍说明。在 2.2 节中，主要介绍了异质信息网络的相关内容，包括 2.2.1 小节定义了异质信息网络的概念、2.2.2 小节阐述了异质信息网络的网络模式，以及 2.2.3 小节介绍了异质信息网络的网络结构。最后，在 2.3 节，主要介绍网络表示学习的相关概念及技术，其中 2.3.1 小节叙述了网络表示学习的定义，2.3.2 小节概述了网络表示学习的应用及度量任务。

## 第三章 结合微观和宏观动态性的动态网络表示学习算法

网络表示学习旨在将网络中节点嵌入到一个低维表示空间，同时捕获网络结构和特性。虽然很多前景可观的网络表示学习工作已经被提出，但是大多数方法聚焦于静态网络。事实上，时序动态网络是非常常见的，其通常随着时间在微观和宏观层面不断演化。微观动态性细致地描述了网络结构的形成过程，而宏观动态性表示网络规模的演变模式。微观动态性和宏观动态性都是网络演变的重要因素。但是，如何准确地在网络表示学习中捕获两者，尤其是宏观动态性，还没有被很好的研究。因此，在本章节中，提出了一种新颖的时序动态网络表示学习方法，其建模并捕获了微观动态性和宏观动态性，称之为 M<sup>2</sup>DNE。

### 3.1 研究背景

网络表示学习利用隐藏表示编码网络结构和特性，为网络分析提供了新颖的角度<sup>[18-20]</sup>。虽然最新一些方法在很多数据挖掘任务上取得了前景可观的表现，但是大多数仅仅考虑了固定结构的静态网络。在实际中，网络往往表现出复杂的时序特性，这意味着网络结构的形成不是一蹴而就的，而通常随着时间而不断演变。在所谓的时序动态网络<sup>[73]</sup>中，边的建立是按时间顺序排列的，并且网络规模也是随着时间以某种显著的分布不断增长的。举例来说，在不同年份，研究者和他人合作，这会导致序列化的合作事件和不断增长的网络规模。因此，一个时序动态网络自然地表示出了网络的演变，不仅包括细粒度的网络结构的演变，还包括宏观的网络规模的变化。用隐空间嵌入时序动态网络，对于实际应用是非常重要的。

基本地，时序动态网络表示学习的一个需求是学习到的网络表示应该保持网络结构并反映网络时序演变过程。时序动态网络的演变通常服从两种动态过程：微观动态性和宏观动态性。在微观层面，时序动态网络的结构由网络中边的建立而驱动，其实际上是一个按时间顺序排列的事件序列，且每个事件由两个节点参与。以图 3-1 为例，从  $t_i$  时刻到  $t_j$  时刻，网络结构的形成可以描述为  $\{(v_3, v_1, t_i), (v_3, v_2, t_i), \dots\} \rightarrow \{(v_3, v_1, t_j), (v_3, v_4, t_j), \dots\}$ ，这里的节点  $v_3$  和  $v_1$  在  $t_j$  时刻再次建立了连接。通常地，在  $t$  时刻一条边的形成必然会和  $t$  时刻之前的历史邻居结构有关，而且邻居结构对当前边的建立的影响来自两方面的节点，而非仅仅一侧。此外，不同的邻居也会有不同的影响。例如，在图 3-1 中，在  $t_j$  时刻边  $(v_3, v_4)$  的建立应该受到  $\{(v_3, v_1, t_i), (v_3, v_2, t_i), \dots\}$  和  $\{(v_4, v_2, t_i), \dots\}$  的影响。此外，由于节点  $v_2, v_3$  和  $v_4$  构成了一个闭合三角，所以  $(v_3, v_2, t_i)$  和  $(v_4, v_2, t_i)$  对事件

$(v_3, v_4, t_j)$  的影响应该比  $(v_3, v_1, t_i)$  的更大。如此微观动态的演变过程，细致地描述了在不同时刻节点间边的形成过程，并且解释了“为何网络在  $t$  时刻会演变为如此结构”。建模这种微观动态性，使得学到的节点表示更准确地捕获时序动态网络的演变，这对于下游的时序动态网络任务是很有益的。在现有工作中，有一些研究者针对时序动态网络表示学习做了研究<sup>[32,34,65]</sup>，但是，他们要么简化网络演变过程为一系列的网络快照，无法真正地反映网络中边的形成顺序<sup>[32,65]</sup>；要么从直接利用随机过程单侧建模邻居结构，忽略了细粒度的结构和时序特性<sup>[34]</sup>。

更重要的，在宏观层面，时序动态网络另一个显著的特性是：随时间推移，网络规模通常表现出明显的变化分布，例如 S-形 sigmoid 曲线或幂律形模式。如图 3-1 所示，当网络随时间演变，网络中的边持续不断地被建立，并在每个时刻形成网络结构。因此，网络规模，即边的数量，随时间增长且服从一定的内在准则，而非随机生成的。这样的宏观动态性反映了时序动态网络的内在演变模式，并在更高的结构层次对网络表示学习施以约束。也就是说，宏观动态性决定了：由微观动态性嵌入而新形成的边的数量。宏观动态性的融入，提供了有价值且有效的演变信息，从而可以提升网络表示学习保持网络结构和演变模式的能力，也会大大增强网络表示学习的泛化能力。因此，学习到的表示空间能否编码宏观动态性应该是时序动态网络表示学习的一个重要要求。遗憾的是，现有的方法没有一个考虑了宏观动态性，即使宏观动态性和时序动态网络演变是非常相关的。

为了实现两个基本要求，本文提出了一种新颖的融合微观和宏观动态性的时序动态网络表示学习方法 M<sup>2</sup>DNE。具体地，为了建模时序动态网络中边的建立过程，即按时间顺序排列的事件（也就是微观动态性），模型 M<sup>2</sup>DNE 通过用网络表示学习表示参数化条件强度函数，设计了一个时序注意力点过程，其利用层级时序注意力捕获细粒度的结构和时序特性。为了建模时序动态网络规模的演变模式（即宏观动态性），模型 M<sup>2</sup>DNE 定义一个由网络嵌入表示参数化的通用动力学方程，其在更高的结构层次对网络表示学习施以约束，并且耦合了时序动态网络中的动力学分析和表示学习。最后，M<sup>2</sup>DNE 整合微观和宏观动态性，并且联合优化两者。由于微观和宏观动态性交互地演变且迭代影响节点表示的学习过程，M<sup>2</sup>DNE 能够统一地捕获拓扑结构的形成过程和网络规模的演变模式。

综上所述，本研究算法的主要贡献可以总结如下：

- 首次研究了结合微观和宏观动态性的动态网络表示学习的重要问题。
- 提出了一个新颖的时序动态网络表示学习模型（M<sup>2</sup>DNE），利用时序注意力点过程微观地建模网络结构的形成，并利用动力学方程宏观地约束网络规模。
- 在三个数据集上进行了丰富的实验，并验证了模型 M<sup>2</sup>DNE 在传统应用（如网络重构）和时序动态网络相关的应用（如规模预测）上的有效性。

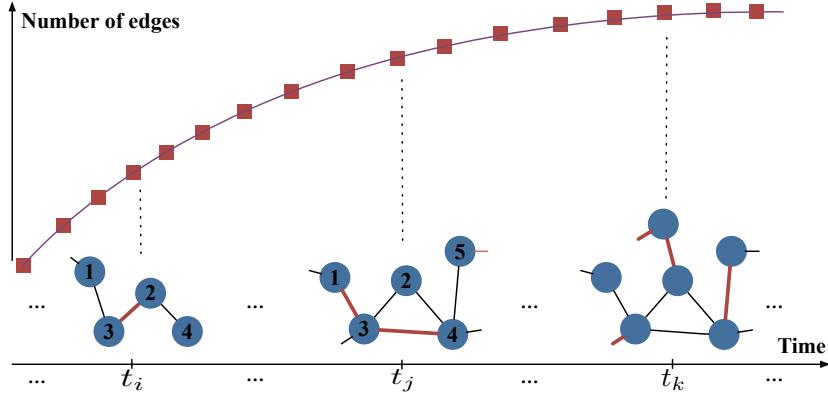


图 3-1 时序动态网络中的微观与宏观动态性。微观动态性描述网络结构的形成(即边建立),而宏观动态性表示网络规模演变模式(即边数量)。红线表示新的时序边(例如:  $(v_3, v_4, t_j)$ )。

## 3.2 算法设计与实现

### 3.2.1 符号及定义

#### 定义 3.1: 时序动态网络

时序动态网络表示一个带有时间戳的边的序列, 其中每条边在某一特定时间连接两个节点。形式化地, 时序动态网络记为  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{T})$ , 其中  $\mathcal{V}$  和  $\mathcal{E}$  分别表示节点和边的集合,  $\mathcal{T}$  是时间戳序列。每条边  $(i, j, t) \in \mathcal{E}$  表示一个由节点  $i$  和  $j$  参与的发生在  $t$  时刻的事件。需要注意的是, 在不同时刻, 节点  $i$  和  $j$  可能建立多条边, 本论文认为  $(i, j, t)$  是一条不同的时序边(即时序交互事件)而  $(i, j)$  表示一条静态边。

#### 定义 2: 微观动态性

给定一个时序动态网络  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{T})$ , 微观动态性描述了网络结构的形成过程, 记为  $\mathcal{I} = \{(i, j, t)_m\}_{m=1}^{|\mathcal{E}|}$ , 其中  $(i, j, t)$  表示一个时序事件, 即节点  $i$  和  $j$  在  $t$  时刻建立边连接。 $\mathcal{I}$  是整个观测到的长度为  $|\mathcal{E}|$  的按时间顺序排列的事件序列。

#### 定义 3: 宏观动态性

给定一个时序动态网络  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{T})$ , 宏观动态性表示网络规模的演变过程, 记为  $\mathcal{A} = \{e(t)\}_{t=t_1}^{|\mathcal{T}|}$ , 其中  $e(t)$  表示到  $t$  时刻为止, 网络中边的数量。

事实上, 宏观动态性既表示节点的变换又表示边的变化。由于新节点的出现必然会导致新边的产生, 因此这里本论文聚焦于边的增长。直观上, 微观动态性决定哪些边将会被建立(即事件的发生), 而宏观动态性约束了新边的规模。

#### 定义 3: 时序动态网络表示学习

给定一个时序动态网络  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{T})$ , 时序动态网络表示学习旨在学习一个映射函数  $f: \mathcal{V} \rightarrow \mathbb{R}^d$ , 其中  $d$  是嵌入表示的维度且  $d \ll |\mathcal{V}|$ 。函数  $f$  的目标是建模网络的演变模式, 包括时序动态网络中的微观和宏观动态性。

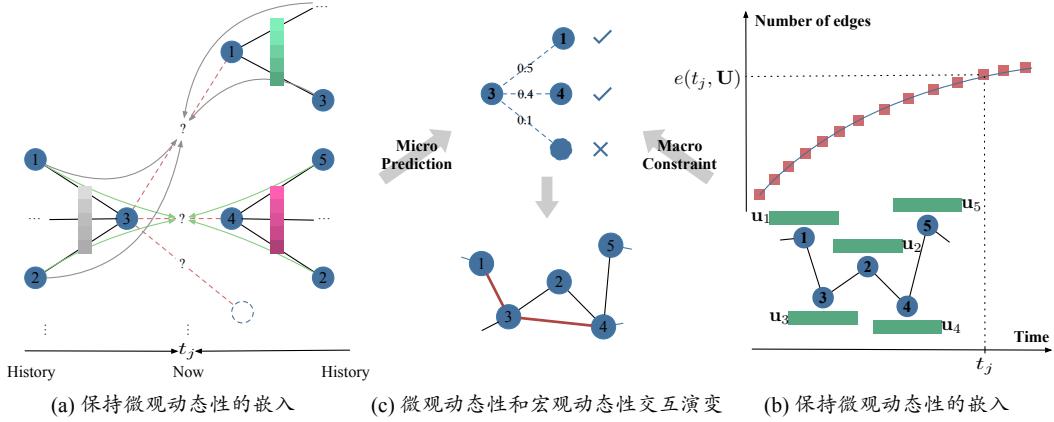


图 3-2 模型 M<sup>2</sup>DNE 总览。(a) 时序注意力点过程保持微观动态性嵌入。虚线表示将要建立的边，带尖头的不同颜色的实线表示邻居对不同边的影响。垂直的彩色矩形块表示注意力机制，颜色越深，表示影响越大。(b) 动力学方程保持宏观动态性嵌入，其由网络表示  $\mathbf{U}$  和时间  $t$  参数化。在  $t_j$  时，M<sup>2</sup>DNE 宏观约束边的数量为  $e(t_j, \mathbf{U})$ 。(c) 微观和宏观动态性相互演变并派生节点表示。在当前时刻  $t_j$ ，依据历史邻居  $\{v_1, v_2, \dots\}$  和  $\{v_5, v_2, \dots\}$ ，模型微观预测节点  $v_3$  将会和节点  $v_1, v_4$  和  $v_2$  分别以 0.5, 0.4 和 0.1 的概率建立连接；而根据网络规模的演变模式，宏观动态性约束新边的数量只有两条。最终，M<sup>2</sup>DNE 可以捕获更加准确的结构和时序特性。

### 3.2.2 算法框架

不同于传统模型只考虑网络结构的演变，本算法融合微观和宏观动态性到时序动态网络表示学习中。如图 3-2 所示，从微观角度（即图 3-2(a)），本论文将网络中边的建立看作是按时间顺序排列的事件，并且提出了一个时序注意力点过程来捕获细粒度的结构和时序特性。边的建立（例如： $(v_3, v_4, t_j)$ ）是由节点本身和其历史邻居（例如： $\{v_1, v_2, \dots\}$  和  $\{v_5, v_2, \dots\}$ ）影响的，其中层级时序注意力捕获邻居的不同影响。从宏观角度（即图 3-2(b)），网络规模的内在演变模式在更高层级约束网络结构，其被定义为一个由网络表示学习表示  $\mathbf{U}$  和时间  $t$  参数化的动力学方程。微观和宏观动态性交替演变并产生节点表示（即图 3-2(c)）。根据历史结构的微观预测，在  $t_j$  时刻，节点  $v_3$  可能会和三个节点建立连接（即三条新边将被建立），而根据网络规模的演变模式，宏观动态性约束了新边的数量为两条。因此由 M<sup>2</sup>DNE 学习得到网络表示能够捕获更准确的结构和时序特性。

### 3.2.3 保持微观动态性的嵌入

随着网络的演变，网络中新边不断被建立，这可以看作是一系列观测到的事件。直观上，事件的发生不仅仅受到事件参与者的影响，还会受到历史事件的影响。此外，历史事件影响当前事件的强度因事件不同而不同。因此，本论文提出了一个时序注意力点过程来保持时序动态网络中的微观动态性。

给定一条时序边 $o = (i, j, t)$ （即一个观测到的事件），利用网络表示学习表示 $\mathbf{U} = [\mathbf{u}]^\top$ 参数化事件强度函数 $\tilde{\lambda}_{i,j}(t)$ 。由于相似的两个节点*i*和*j*更有可能建立边 $(i, j, t)$ ，因此节点之间的相似性应该正比于*i*和*j*在*t*时刻建立连接这一事件的强度。另一方面，历史邻居和当前节点的相似性表示历史事件对当前事件 $(i, j, t)$ 的影响强度，其应该随时间衰减且因邻居不同而不同。为此，事件 $o = (i, j, t)$ 的发生强度可以定义如下，包括来自节点本身的基础强度和来自两侧邻居额度历史影响：

$$\begin{aligned}\tilde{\lambda}_{i,j}(t) &= \underbrace{g(\mathbf{u}_i, \mathbf{u}_j)}_{\text{Base Intensity}} \\ &+ \beta_{ij} \sum_{p \in \mathcal{H}^i(t)} \alpha_{pi}(t) g(\mathbf{u}_p, \mathbf{u}_j) \kappa(t - t_p) \\ &+ (1 - \beta_{ij}) \underbrace{\sum_{q \in \mathcal{H}^j(t)} \alpha_{qj}(t) g(\mathbf{u}_q, \mathbf{u}_i) \kappa(t - t_q)}_{\text{Neighbor Influence}},\end{aligned}\quad (3-1)$$

其中 $g(\cdot)$ 是度量节点相似性的函数，这里定义为 $g(\mathbf{u}_i, \mathbf{u}_j) = -\|\mathbf{u}_i - \mathbf{u}_j\|_2^2$ ，其他相似性度量方法也可以用于此处，例如余弦相似性。 $\mathcal{H}^i(t)$ 和 $\mathcal{H}^j(t)$ 分别是节点*i*和*j*在*t*时刻之前的历史邻居。 $\kappa(t - t_p) = \exp(-\delta_i(t - t_p))$ 项是时间衰减函数，可学习的参数 $\delta_i > 0$ 是节点依赖的， $t_p$ 是历史事件 $(i, p, t_p)$ 发生的时间。这里的 $\alpha$ 和 $\beta$ 是两个注意力参数，由一个层次时序注意力机制定义，后文会进行详细介绍。

由于当前事件是由历史事件随机激活或抑制的，并且公式(3-1)可能会导致负数值的出现，本算法进一步施加一个非线性转换函数： $f: \mathbb{R} \rightarrow \mathbb{R}_+$ （即：指指数级函数）来确保事件的发生强度是正实数值。因此，有如下公式：

$$\lambda_{i,j}(t) = f(\tilde{\lambda}_{i,j}(t)) \quad (3-2)$$

### 3.2.3.1 层次时序注意力机制

如前文所述，历史事件会对当前事件的发生产生影响，且影响会因历史事件不同而不同。例如，在*t*时刻，研究者*i*和*j*是否会合作一篇神经网络相关的论文通常和他们各自的历史合作者有关。直观上，和*i*或者*j*有过关于神经网络相关论文合作的研究者会对当前事件 $(i, j, t)$ 有较大的局部影响。此外，如果*i*的历史合作者在整体上更专业于神经网络，那么其的历史邻居会有一个更大的全局影响。由于研究者的研究方向会随着研究热点的出现而变化，所以邻居的影响也是动态变化的。因此，本算法提出一个时序层次注意力来捕获非均匀且动态的历史结构影响。

对于来自每个邻居的局部影响， $g(p, j) = -\|\mathbf{u}_p - \mathbf{u}_j\|_2^2$ 使得两个节点*i*和*j*更有可能建立一条连接，如果*i*的邻居*p*是和*j*很相似的。*p*对于事件 $(i, j, t)$ 的重要性取决于当前节点*i*，并且随着邻居结构的变化而变化。因此，注意力系数定义为：

$$\tilde{\alpha}_{pi}(t) = \sigma(\kappa(t - t_p) \mathbf{a}^\top [\mathbf{Wu}_i \oplus \mathbf{Wu}_p]) \quad (3-3)$$

$$\alpha_{pi}(t) = \frac{\exp(\tilde{\alpha}_{pi}(t))}{\sum_{p' \in \mathcal{H}^i(t)} \exp(\tilde{\alpha}_{p'i}(t))} \quad (3-4)$$

其中  $\mathbf{a} \in \mathbb{R}^{2d}$  充当注意力向量且  $\mathbf{W}$  表示局部权重矩阵，这里整合了时间衰减函数  $\kappa(t - t_p)$ ，以此如果时间戳  $t_p$  是接近  $t$  的，那么节点  $p$  会对事件  $o$  有较大的影响。同样地，可以得到  $\alpha_{qj}(t)$ ，其捕获节点  $j$  的历史邻居对当前事件  $o$  在  $t$  时刻的局部影响。

对于整体邻居的全局影响，首先将历史邻居表示为一个整体，即每个邻居信息的聚合  $\tilde{\mathbf{u}}_i = \sigma(\sum_{p \in \mathcal{H}^i(t)} \alpha_{pi}(t) \mathbf{W} \mathbf{u}_i)$ 。考虑到全局时间衰减，均值化时间衰减  $\overline{t - t_p} = \frac{1}{|\mathcal{H}^i(t)|} \sum_{p \in \mathcal{H}^i(t)} (t - t_p)$ 。因此，整体邻居对当前事件发生的全局注意力：

$$\beta_{ij} = \frac{\exp(s(\kappa(\overline{t - t_p}) \tilde{\mathbf{u}}_i))}{\exp(s(\kappa(\overline{t - t_p}) \tilde{\mathbf{u}}_i)) + \exp(s(\kappa(\overline{t - t_q}) \tilde{\mathbf{u}}_j))} \quad (3-5)$$

其中  $s(\cdot)$  是一个单层神经网络，其接受邻居的聚合表示  $\tilde{\mathbf{u}}_i$  和均值化的时间衰减  $\kappa(\overline{t - t_p}) = \exp(-\delta_i(\overline{t - t_p}))$  为输入。

联合两部分注意力，算法可以以耦合的方式保持结构和时序特性，因为注意力本身是随着时序动态网络中的微观动态性演变而变化的。

### 3.2.3.2 微观预测

至此，算法定义节点  $i$  和  $j$  在  $t$  时刻建立一条边的概率为：

$$p(i, j | \mathcal{H}^i(t), \mathcal{H}^j(t)) = \frac{\lambda_{i,j}(t)}{\sum_{i' \in \mathcal{H}^j(t)} \lambda_{i',j}(t) + \sum_{j' \in \mathcal{H}^i(t)} \lambda_{i,j'}(t)} \quad (3-6)$$

因此，算法最小化如下目标函数来捕获时序动态网络中的微观动态性：

$$\mathcal{L}_{mi} = - \sum_{t \in \mathcal{T}, (i,j,t) \in \mathcal{E}} \log p(i, j | \mathcal{H}^i(t), \mathcal{H}^j(t)) \quad (3-7)$$

### 3.2.4 保持宏观动态性的嵌入

区别于微观动态性驱动网络中边的形成，宏观动态性描述了网络规模的演变，其通常服从显著的分布，即网络规模可以被刻画为特定的动力学方程。更进一步地，宏观动态性在更高层次约束了网络结构的形成，即其决定了到目前为止新产生边的数量。编码这种高层次的结构特性能够大大增强网络表示学习的能力。因此，本论文提出定义一个由网络表示学习表示参数化的动力学方程，其在时序动态网络的动力学分析和表示学习之间建立了桥梁。

给定一个时序动态网络  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{T})$ ，可以得到在  $t$  时刻的节点数量  $n(t)$ 。对于每一个节点  $i$ ，其以  $r(t)$  的连接率在  $t$  时刻连接其他节点（例如：节点  $j$ ）。根据网络演变中的致密幂律，可以得到节点  $i$  的平均可达邻居数量  $\zeta(n(t) - 1)^\gamma$ ，其中  $\zeta$  表示线性稀疏系数， $\gamma$  表示幂律稀疏指数。因此本算法定义宏观动态性如下：

$$\Delta e'(t) = n(t)r(t)(\zeta(n(t) - 1)^\gamma) \quad (3-8)$$

其中随着网络演变到  $t$  时刻， $n(t)$  可以获得，且  $\zeta$  和  $\gamma$  随着模型优化是可学习的。

当网络演变到 $t$ 时刻,  $n(t)$ 个节点加入到网络中。在下一时刻, 网络中的每个节点尝试和其他 $\zeta(n(t) - 1)^{\gamma}$ 个节点以 $r(t)$ 的连接率建立连接。

### 3.2.4.1 连接率

由于连接率 $r(t)$ 在驱动网络规模演变过程中扮演重要角色, 连接率不仅仅依赖于时序信息, 还取决于网络的结构信息。一方面, 网络中非常多的边时在网络演变初期建立的, 而随着网络的致密演变增长速度不断衰减。因此, 连接率应该随着某一时序项不断衰减。另一方面, 网络中边的建立, 推动网络结构的演变, 因此连接率应该和网络的结构特性有关。因此, 为了在网络表示学习表示中捕获这种时序和网络信息, 算法利用时序项和节点表示参数化连接率:

$$r(t) = \frac{\frac{1}{|\mathcal{E}|} \sum_{(i,j,t) \in \mathcal{E}} \sigma(-\|\mathbf{u}_i - \mathbf{u}_j\|_2^2)}{t^\theta} \quad (3-9)$$

其中 $\theta$ 是时序项指数,  $\sigma(x)$ 是 sigmoid 函数。由于模型学习到的节点表示应该很好地编码网络结构, 上式分子项建模了网络的最大连接率, 并随时间衰减。

### 3.2.4.2 宏观约束

随着网络演变, 有网络中边数量的序列 $\mathcal{A} = \{e(t)\}_{t=t_1}^{|\mathcal{T}|}$ , 因此可以得到网络中边的变化量, 记为 $\{\Delta e(t_1), \Delta e(t_2), \Delta e(t_3), \dots\}$ , 其中 $\Delta e(t_i) = e(t_{i+1}) - e(t_i)$ 。接下来学习公式(3-10)中的参数, 通过最小化如下平方误差和:

$$\mathcal{L}_{ma} = \sum_{t \in \mathcal{T}} (\Delta e(t) - \Delta e'(t))^2 \quad (3-10)$$

## 3.2.5 模型优化

由于微观和宏观动态性相互驱动动态网络的演变, 其交替影响网络表示学习的过程, 以统一的方式捕获网络的拓扑结构的形成过程和网络规模的演变模式:

$$\mathcal{L} = \mathcal{L}_{mi} + \epsilon \mathcal{L}_{ma} \quad (3-11)$$

其中 $\epsilon \in [0,1]$ 是宏观动态性对网络表示学习的权重。

由于 $\mathcal{L}_{ma}$ 是一个非线性最小平方问题, 可以用梯度下降解决。但 $\mathcal{L}_{mi}$ 的计算是复杂的, 因为 $p(i, j | \mathcal{H}^i(t), \mathcal{H}^j(t))$ 的计算是复杂耗时的。为了解决这一问题, 可以将公式(3-2)中的转换函数 $f$ 定义为 $\exp(\cdot)$ 函数, 因此公式(3-7)实际上是一个施加在 $\tilde{\lambda}_{i,j}(t)$ 上的 Softmax 项, 其可以通过负采样近似优化。具体的, 算法 M<sup>2</sup>DNE 每次以正比于边权重的概率采样一个正样本边 $(i, j, t) \in \mathcal{E}$ , 然后采样 $K$ 个负样本对 $(i', j, t)$ 和 $(i, j', t)$ 。因此, 公式(3-8)的损失函数可以重写作:

$$\begin{aligned} \mathcal{L}_{mi} = & - \sum_{t \in \mathcal{T}} \sum_{(i,j,t) \in \mathcal{E}} \log \sigma(\tilde{\lambda}_{ij}(t)) \\ & - \sum_{k=1}^K \mathbb{E}_{i'} [\log (\sigma(-\tilde{\lambda}_{i'j}(t)))] - \sum_{k=1}^K \mathbb{E}_{j'} [\log (\sigma(-\tilde{\lambda}_{ij'}(t)))] \end{aligned} \quad (3-12)$$

其中 $\sigma(x) = \exp(x)/(1 + \exp(x))$ 是 sigmoid 函数。需要注意的是，M<sup>2</sup>DNE 固定最大长度的历史邻居序列 $h$ 并且保留最新的邻居节点。

M<sup>2</sup>DNE 的时间复杂度是 $O(I\mathcal{T}(dhK|\mathcal{V}| + |\mathcal{E}|))$ ，其中 $I$ 是迭代次数， $\mathcal{T}$ 是时间戳的数量。 $d$ 是节点表示维度， $h$ 和 $K$ 分别是邻居和负样本的数量。

### 3.3 实验及分析

#### 3.3.1 实验设置

##### 3.3.1.1 数据集

本实验采用来自不同领域的三个数据集，即 Eucore, DBLP 和 Tmall。Eucore 是基于电子邮件数据生成的，其中人与人之间的沟通表示边连接，五个工作部门被视为标签。DBLP 是一个学术合作网络，十个研究领域作为标签。Tmall 是基于电商销售数据抽取的，以用户和物品为节点，以购买行为为边连接，其中购买最多的商品的五个类别作为标签。表 3-1 总结了数据集的详细统计信息。

表 3-1 数据集统计信息

Datasets	Eucore	DBLP	Tmall
# nodes	986	28,085	577,314
# static edges	24,929	162,451	2,992,964
# temporal edges	332,334	236,894	4,807,545
# time steps	526	27	186
# labels	5	10	5

##### 3.3.1.2 对比方法

为了验证 M<sup>2</sup>DNE 的有效性，本论文对比了 7 个算法，其中包括：针对静态同质网络表示学习的 DeepWalk<sup>[24]</sup>、node2vec<sup>[25]</sup>、LINE<sup>[46]</sup>和 SDNE<sup>[26]</sup>；针对动态同质网络的 TNE、DynamicTriad 和 HTNE<sup>[34]</sup>。此外，作为 M<sup>2</sup>DNE 的变种模型，MDNE 仅建模微观动态性，用以验证宏观动态性对动态网络表示学习的重要性。

##### 3.3.1.3 参数设置

本实验中所有方法的节点向量维度为 128，负样本数量为 5。对于 M<sup>2</sup>DNE，在 Eucore, DBLP 和 Tmall 上，历史邻居长度分别为 2, 5 和 2，平衡系数 $\epsilon$ 分别为 0.3, 0.4 和 0.3。M<sup>2</sup>DNE 采用小批量随机梯度下降优化，并用 PyTorch 实现。

### 3.3.2 有效性实验

#### 3.3.2.1 网络重构

在这个任务中，算法 M<sup>2</sup>DNE 在完全演变结束的网络上学习节点的表示，并根据节点之间的相似度重构边连接。根据已有工作<sup>[74]</sup>，对于 DeepWalk, node2vec,

LINE 和 TNE，由于其模型本身的基于内积的优化目标，我们将节点表示向量之间的内积作为相似性。对于 SDNE, DynamicTriad, HTNE 和本文的模型(即 MDNE 和 M<sup>2</sup>DNE)，采用负平方的欧几里得距离作为节点相似性。然后，根据相似性对节点对进行排序。由于在 DBLP 和 Tmall 数据集中可能的节点对的数量（即  $|\mathcal{V}|(|\mathcal{V}| - 1)/2$ ）过大，本实验参考已有工作<sup>[45]</sup>随机选取 1% 和 0.1% 的节点对进行评估，并以 Precision@K 和 AUC 作为评价指标。

表 3-2 网络重构实验结果

Methods	Eucore			DBLP			Tmall		
	Pre.@100	Pre.@1000	AUC	Pre.@100	Pre.@1000	AUC	Pre.@100	Pre.@1000	AUC
DeepWalk	0.56	0.576	0.7737	0.92	0.321	0.9617	0.55	0.455	0.8852
node2vec	0.73	0.535	0.8157	0.81	0.248	0.9833	<b>0.58</b>	<b>0.493</b>	0.9755
LINE	0.58	0.487	0.7711	0.89	0.524	0.9859	0.11	0.183	0.8355
SDNE	0.77	0.747	0.8833	0.88	0.278	0.8945	0.24	0.387	0.8934
TNE	0.89	0.778	0.6803	0.03	0.013	0.9003	0.01	0.062	0.7278
DynamicTriad	0.91	0.745	0.7234	0.93	0.469	0.7464	0.27	0.324	0.9534
HTNE	0.84	0.776	0.9215	0.95	0.528	0.9944	0.40	0.404	0.9804
MDNE	0.94	0.809	0.9217	0.97	0.543	0.9953	0.25	0.412	0.9853
M <sup>2</sup> DNE	<b>0.96</b>	<b>0.823</b>	<b>0.9276</b>	<b>0.99</b>	<b>0.553</b>	<b>0.9964</b>	0.30	0.431	<b>0.9865</b>

表 3-2 表明，在 AUC 指标上，本文提出的 MDNE 和 M<sup>2</sup>DNE 始终优于所有的对比方法。在 Precision@K 指标上，M<sup>2</sup>DNE 在 Eucore 和 DBLP 上取得了最佳性能，相较于最优对比方法，分别有 5.78% 和 4.73% 的提升。如此显著的提升的得益于 M<sup>2</sup>DNE 模型建模了宏观动态性，其在网络规模上宏观约束了噪声边关系的建立，从而准确的重构了网络中的真实边连接。虽然 MDNE 模型仅对微观动力学进行了建模，但是它通过时序注意力点过程捕获了细粒度的结构特性，因此在 Eucore 和 DBLP 数据集上仍然优于所有的对比方法。M<sup>2</sup>DNE 方法在 Tmall 数据集上表现不佳的一个潜在原因是，短期内购买行为的演化模式不明显，这导致时态网络表示学习模型(即 TNE, DynamicTriad, HTNE, MDNE 和 M<sup>2</sup>DNE)都无法很好地建模网络演变趋势，导致表现不佳。然而，在大多数情况下，M<sup>2</sup>DNE 仍然比其他时态网络表示学习方法表现更好，这表明需要共同捕获微观和宏观动态性以进行时态动态网络嵌入。

### 3.3.2.2 节点分类

在完全演变结束的网络上学习节点的表示之后，本实验训练一个逻辑回归分类器，该分类器将节点嵌入表示作为输入特征。训练集的比率设置为 40%，60% 和 80%，并在表 3-3 中报告 Macro-F1 和 Micro-F1 指标结果。

正如所观察到的，绝大多数情况下，所提出的 MDNE 和 M<sup>2</sup>DNE 都取得了优于对比方法的分类结果。具体而言，与静态网络表示学习方法(如 DeepWalk, node2vec, LINE 和 SDNE)相比，MDNE 和 M<sup>2</sup>DNE 的良好性能表明，模型中保留的网络结构的形成过程提供了有效的信息，可以使节点的表示向量更具区分性。

在时序动态网络表示学习方法（即 TNE, DynamicTriad 和 HTNE）方面，MDNE 和 M<sup>2</sup>DNE 的通过分层的时态注意力机制捕获了邻居聚集的局部和全局结构，从而增强了结构嵌入的准确性。此外，M<sup>2</sup>DNE 在潜在嵌入空间中对高阶结构进行编码，从而进一步提高了分类性能。纵向比较，在大多数情况下，MDNE 和 M<sup>2</sup>DNE 在不同大小的训练集上仍然表现最佳，表明了模型的稳定性和鲁棒性。

表 3-3 节点分类实验结果

Datasets	Metrics	Tr.Ratio	DeepWalk	node2vec	LINE	SDNE	TNE	DynamicTriad	HTNE	MDNE	M <sup>2</sup> DNE
Eucore	Macro-F1	40%	<b>0.1878</b>	0.1575	0.1765	0.1723	0.0954	0.1486	0.1319	0.1598	0.1365
		60%	0.1934	0.1869	0.1777	0.1834	0.1272	0.1796	0.1731	0.1855	<b>0.1952</b>
		80%	0.2049	0.2022	0.1278	0.1987	0.1389	0.1979	0.1927	0.1948	<b>0.2057</b>
	Micro-F1	40%	0.2089	0.2133	0.2266	0.2129	0.2298	0.2310	0.2200	0.2273	<b>0.2311</b>
		60%	0.2245	0.2400	0.1933	0.2321	0.2377	0.2333	0.2400	0.2501	<b>0.2533</b>
		80%	0.2400	0.2660	0.1466	0.2543	0.2432	0.2400	0.2672	0.2702	<b>0.2800</b>
DBLP	Macro-F1	40%	0.6708	0.6607	0.6393	0.5225	0.0580	0.6045	0.6768	0.6883	<b>0.6902</b>
		60%	0.6717	0.6681	0.6499	0.5498	0.1429	0.6477	0.6824	0.6915	<b>0.6948</b>
		80%	0.6712	0.6693	0.6513	0.5998	0.1488	0.6642	0.6836	0.6905	<b>0.6975</b>
	Micro-F1	40%	0.6653	0.6680	0.6437	0.5517	0.2872	0.6513	0.6853	0.6892	<b>0.6923</b>
		60%	0.6689	0.6737	0.6507	0.5932	0.2931	0.6680	0.6857	0.6922	<b>0.6947</b>
		80%	0.6638	0.6731	0.6474	0.6423	0.2951	0.6695	0.6879	0.6924	<b>0.6971</b>
Tmall	Macro-F1	40%	0.4909	0.5437	0.4371	0.4845	0.1069	0.4498	0.5481	0.5648	<b>0.5775</b>
		60%	0.4929	0.5455	0.4376	0.4989	0.1067	0.4897	0.5489	0.5681	<b>0.5799</b>
		80%	0.4953	0.5458	0.4397	0.5312	0.1068	0.5116	0.5493	0.5728	<b>0.5847</b>
	Micro-F1	40%	0.5711	0.6041	0.5367	0.5734	0.3647	0.5324	0.6253	0.6344	<b>0.6421</b>
		60%	0.5734	0.6056	0.5392	0.5788	0.3638	0.5688	0.6259	0.6369	<b>0.6438</b>
		80%	0.5778	0.6066	0.5428	0.5832	0.3642	0.6072	0.6264	0.6401	<b>0.6465</b>

### 3.3.3 微观动态性实验

通过时序节点推荐和链路预测任务，本实验验证模型 M<sup>2</sup>DNE 对微观动态性的建模能力。由于网络演化是一个长期的过程，而 Tmall 的边连接具有不太明显的演化模式，且实验结果较差，因此这里在 Eucore 和 DBLP 数据集上进行了实验。具体来说，给定一个测试时间戳  $t$ ，在时间  $t$ （不包括在内）之前的网络上学习节点的嵌入表示，并在时间  $t$ （包括在内）之后的网络上进行评估预测性能。对于 Eucore 数据，由于进化时间长，本实验将前 500 个时间戳设置为训练数据。对于 DBLP 数据，本实验在前 26 个时间戳之前的网络上学习节点嵌入表示。

#### 3.3.3.1 时序节点推荐

对于  $t$  时刻之前的网络中的每个节点  $v_i$ ，本实验预测  $t$  时刻节点  $v_i$  的 top-K 个可能的邻居节点。类似于网络重构实验，本实验采用相同的相似性度量方式，计算排序分数，然后取排名前 K 个节点作为候选邻居节点。本实验目的在评估模型 M<sup>2</sup>DNE 对时序信息的建模能力，为了提供更全面的比较，这里还对比了一个针对静态网络表示学习的方法（即 DeepWalk）。实验以 Recall@K 和 Precision@K 作为度量指标，图 3-3 汇报了实验结果。

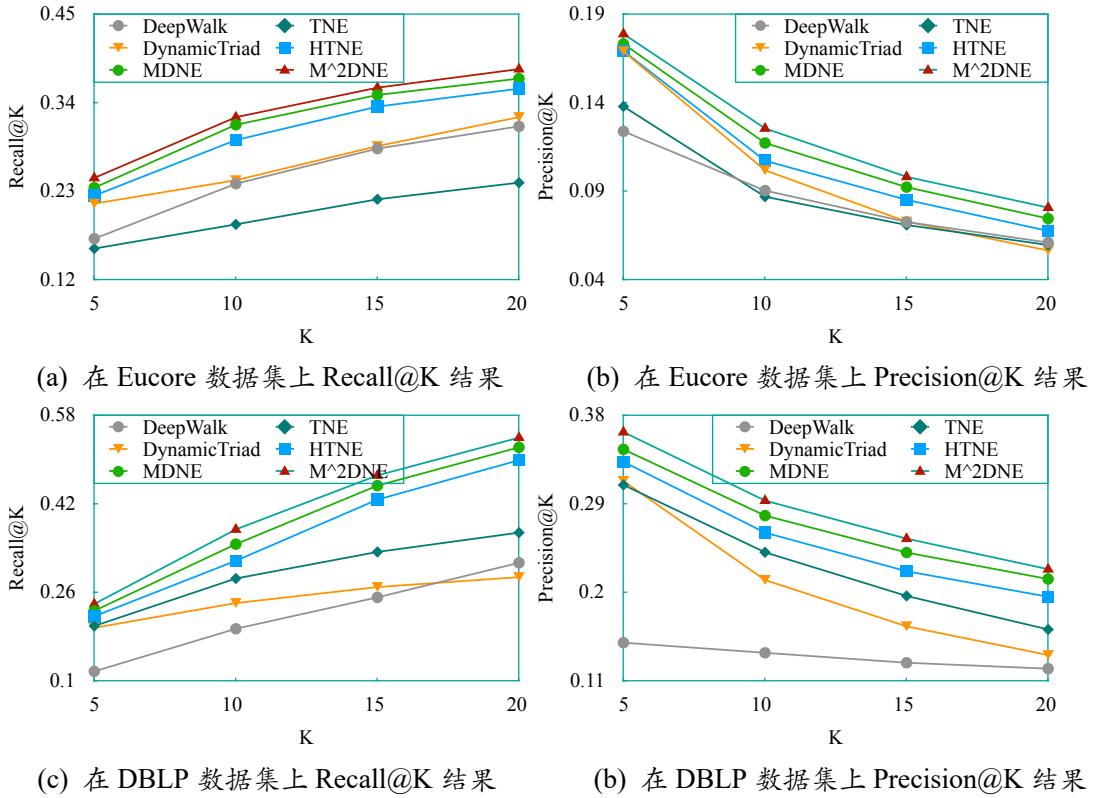


图 3-3 时序节点推荐实验结果

总体来看，在不同指标上，本论文提出的模型 MDNE 和 M<sup>2</sup>DNE 的性能均优于所有对比方法。在 Eucore 数据集上，和最优的对比方法（HTNE）相比，模型 M<sup>2</sup>DNE 的推荐性能分别提升了 10.88% 和 8.34%，就 Recall@10 和 Precision@10 两个指标而言。在 DBLP 数据集上，Recall@10 和 Precision@10 两个指标分别提升了 6.05% 和 11.69%。这些显著的提升验证了模型 MDNE 和 M<sup>2</sup>DNE 中的时序注意力点过程有效的建模了网络的细粒度结构和微观动态演变模式。此外，本论文所提出的模型 M<sup>2</sup>DNE 的显著提升得益于对宏观动态性的建模，其对网络的嵌入表示进行了高阶的约束，从而实现了对网络内在演化方式的编码。

### 3.3.3.2 时序链路预测

在本实验中，模型 M<sup>2</sup>DNE 在  $t$  时刻之前的网络上学习节点的向量表示，并预测  $t$  时刻之后的边连接的建立。给定节点  $i$  和  $j$ ，节点之间的边连接的表示定义为  $|\mathbf{u}_i - \mathbf{u}_j|$ 。本实验将  $t$  时刻构建的边连接视作正样本，并随机采样相同数量的负样本（即两个节点不相连）。基于构建的数据及节点的表示向量，本实验训练一个逻辑回归分类器，以评估模型表现。

如表 3-4 所示，MDNE 和 M<sup>2</sup>DNE 在两个指标上始终优于所有对比方法。可以观察到，TNE 和 DynamicTriad 的性能较差，这是由于他们基于快照动态网络建模网络的演变趋势，这在一定程度上忽略了边连接的形成过程。虽然 HTNE 模型考虑了这种时序动态过程，但该模型忽略了网络规模的演化模式，仅仅建模了

单边邻域结构信息。本论文提出的 M<sup>2</sup>DNE 从微观和宏观两个角度建模网络的动态演变，因此更好地编码了动态时序网络中的时序和结构信息。

表 3-4 时序链路预测实验结果

Methods	Eucore		DBLP	
	ACC.	F1	ACC.	F1
DeepWalk	0.8444	0.8430	0.7776	0.7778
node2vec	0.8591	0.8583	0.8128	0.8059
LINE	0.7837	0.7762	0.6711	0.6756
SDNE	0.7533	0.7908	0.6971	0.6867
TNE	0.6932	0.6691	0.5027	0.4799
DynamicTriad	0.6775	0.6611	0.6189	0.6199
HTNE	0.8539	0.8498	0.8123	0.8157
MDNE	0.8649	0.8585	0.8292	0.8239
M <sup>2</sup> DNE	<b>0.8734</b>	<b>0.8681</b>	<b>0.8336</b>	<b>0.8341</b>

### 3.3.4 宏观动态性实验

#### 3.3.4.1 规模预测

在此任务中，目的在于预测特定时间的网络边数。在 Eucore, DBLP 和 Tmall 三个数据集上，模型 M<sup>2</sup>DNE 分别在前 500、26 和 180 个时间戳构成的网络上进行节点表示学习，然后预测下一时刻的累积边数。对于对比方法，由于他们都无法预测网络中边的数量，因此本实验定义每个节点对的分数为  $\phi(i, j) = \sigma(\mathbf{u}_i \mathbf{u}_j)$ ，如果  $\phi(i, j) > 0.5$ ，则在下一时刻节点  $i$  和  $j$  会建立边连接，否则不建立边连接。对于本论文提出的算法 M<sup>2</sup>DNE，边连接的数量定义为公式(3-9)。

表 3-5 网络规模预测实验结果

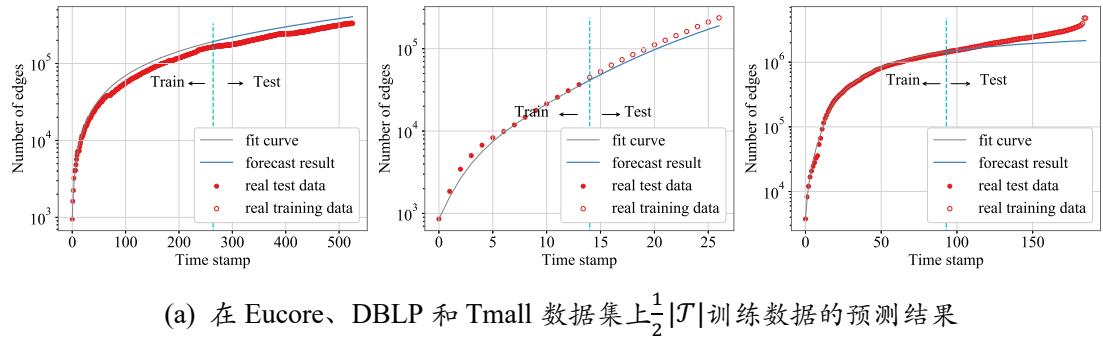
Methods	Eucore			DBLP			Tmall		
	$e'(t)$	$e(t)$	A.E.	$e'(t)$	$e(t)$	A.E.	$e'(t)$	$e(t)$	A.E.
DeepWalk	444,539	24,929	419,610	335,916,746	162,451	335,754,295	118,381,361,880	2,992,964	118,378,368,916
node2vec	479,583	24,929	454,654	363,253,815	162,451	363,091,364	135,349,949,950	2,992,964	135,346,956,986
LINE	278,175	24,929	253,246	363,567,406	162,451	363,404,955	135,763,029,298	2,992,964	135,760,036,334
SDNE	396,752	24,929	371,823	361,748,486	162,451	361,586,035	134,748,693,450	2,992,964	134,745,700,486
TNE	485,584	332,334	153,250	389,257,712	236,894	389,020,818	166,630,196,186	4,807,545	166,625,388,641
DynamicTriad	485,605	332,334	163,271	394,369,570	236,894	394,132,676	165,467,872,223	4,807,545	165,463,064,678
HTNE	203,012	332,334	129,322	173,501,036	236,894	173,264,142	82,716,705,256	4,807,545	82,711,897,711
MDNE	203,776	332,334	128,558	173,205,229	236,894	172,968,335	82,702,894,887	4,807,545	82,698,087,342
M <sup>2</sup> DNE	349,157	332,334	<b>16,823</b>	222,993	236,894	<b>13,901</b>	3,855,548	4,807,545	<b>951,997</b>

以绝对预测误差作为评测指标，表 3-5 汇报了在三个数据集上的实验结果。在这里， $e(t)$  表示边连接的真实数量， $e'(t)$  则表示模型的预测值。对于静态网络表示学习模型，这里只计算静态边连接，即同样的两个节点间的连接只计算一次边连接数量。这也就是为什么表 3-5 中不同方法的  $e(t)$  不同的原因。很显然地，本文提出的模型 M<sup>2</sup>DNE 可以精确地预测动态网络中边连接的数量，这证明了由网络嵌入表示参数话的动力学方程的有效性，其很好的建模了网络规模的演化模

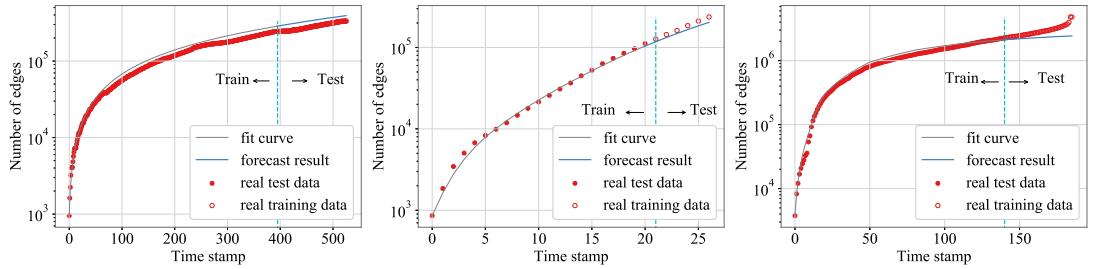
式。相对的，对比方法的预测误差都很大，这是因为这些方法本身是无法建模网络宏观动态演变的，从而无法准确预测网络规模的演变。

### 3.3.4.2 趋势预测

与规模预测不同，此任务旨在预测网络规模的总体演变趋势。给定时间戳  $t$ ，本实验在时间  $t$ （不包括在内）之前的网络上学习节点的嵌入表示（即  $\mathbf{U}$ ）和动力学方程参数（即  $\theta, \zeta$  和  $\gamma$ ）。基于学习到的网络表示和参数，该实验预测了网络规模的演化趋势并绘制曲线。由于所有的对比方法都无法预测网络规模，在这里只报告本论文提出的模型的预测结果。在该实验中，时间戳  $t$  分别设置为  $\frac{1}{2}|\mathcal{T}|$  和  $\frac{3}{4}|\mathcal{T}|$ 。



(a) 在 Eucore、DBLP 和 Tmall 数据集上  $\frac{1}{2}|\mathcal{T}|$  训练数据的预测结果



(b) 在 Eucore、DBLP 和 Tmall 数据集上  $\frac{3}{4}|\mathcal{T}|$  训练数据的预测结果

图 3-4 趋势预测实验结果。红色圆表示真实数据分布，实心圆用于训练，空心圆用于测试。灰色线表示训练数据拟合曲线，蓝色线表示测试数据预测拟合曲线。

如图 3-4 所示，M<sup>2</sup>DNE 可以很好地拟合动态网络中边连接的数量，这证明由网络嵌入表示参数化的连接率准确连接了网络动态性和网络结嵌入表示。此外，随着训练时间戳的增大（即从  $\frac{1}{2}|\mathcal{T}|$  到  $\frac{3}{4}|\mathcal{T}|$ ），预测误差逐渐减小，这也是和常识相符的。因为更多的训练数据有助于学习节点的嵌入表示和演变趋势，从而更好的预测网络的增长模式。同时，我们还注意到，在 Tmall 数据集上，模型表现欠佳，这是因为在短期内购买行为的演化模式不那么显著，从而很难刻画网络的增长。

### 3.3.5 参数分析实验

#### 3.3.5.1 历史邻居数量

如图 3-5(a) 所示，模型 M<sup>2</sup>DNE 的性能比较稳定并在  $h = 5$  时达到最优。由于历史邻居建模了网络结构的形成，因此邻居的数量会影响模型性能。为了在性能和复杂度之间取得平衡，在 M<sup>2</sup>DNE 中，邻居数量被设置为一个较小的数值。

### 3.3.5.2 负采样数量

如图 3-5(b) 所示, 模型 M<sup>2</sup>DNE 的性能随着负样本数量的增加而提高, 然后负样本数量达到 5 的时候达到最佳性能。总的来说, 模型 M<sup>2</sup>DNE 的性能是稳定的, 这证明了模型对负样本的数量较不敏感。

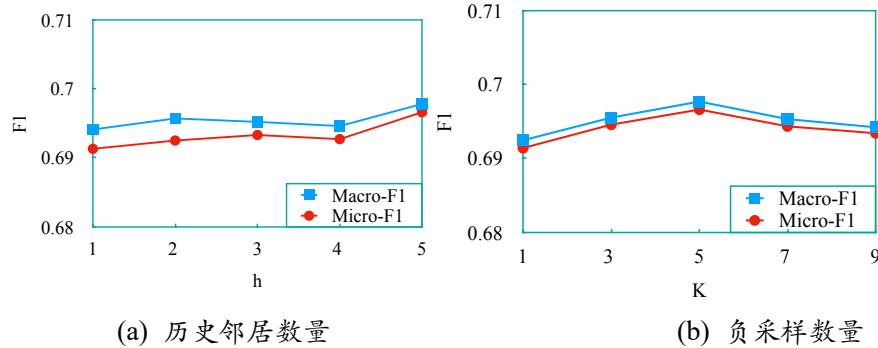


图 3-5 参数实验结果

## 3.4 本章小结

本章介绍了结合微观和宏观演变的动态网络表示学习算法 (M<sup>2</sup>DNE)。在 3.1 节, 介绍了模型 M<sup>2</sup>DNE 的提出背景及创新贡献, 并概述了模型的基本思想。3.2 节详细地叙述了算法设计细节, 其中 3.2.1 小节对相关符号进行了介绍, 其次 3.2.2 小节概述了算法框架, 然后 3.2.3 小节和 3.2.4 小节分别详细介绍了保持微观动态性和宏观动态性的嵌入模块, 最后 3.2.5 小节总结了模型的损失函数和优化方式。在 3.3 节, 主要报告了针对模型 M<sup>2</sup>DNE 的实验及分析, 包括 3.3.1 小节的实验设置, 3.3.2 小节对模型有效性的验证实验, 3.3.3 小节和 3.3.4 小节对模型微观和宏观动态性的验证实验, 以及 3.3.5 小节对模型参数的分析实验。

## 第四章 关系结构感知的异质信息网络表示学习算法

异质信息网络 (HIN) 表示学习旨在将多类型的节点嵌入到一个低维度的向量空间。虽然大多数已有的异质信息网络表示学习方法考虑了网络中的异质性，但是他们通常使用单一的模型不加以区分地建模所有关系，这必然会削弱网络嵌入的能力。在本章节中，考虑了异质关系的结构特征并提出了一个新颖的关系结构感知的异质信息网络表示学习模型 (RHINE)。通过对真实网络数据透彻的数学分析和探索，本章节提出了两种结构相关的度量方法，这两种方法可以将异质的关系一致地划分为两类：隶属关系 (Affiliation Relations, ARs) 和交互关系 (Interaction Relations, IRs)。为了充分建模关系中有区别的特征，本章节提出两种特别设计的模型分别处理 ARs 和 IRs，以此可以保证捕获网络的拓扑结构和语义信息。最后，联合各个模型，并以统一的方式训练优化。在三个真实数据集上的丰富实验证明了模型 RHINE 的有效性。在多个数据挖掘任务中，包括聚类、分类、链路预测等，模型 RHINE 的表现均优于现有的网络表示学习模型。

### 4.1 研究背景

由于网络嵌入可以有效的学习编码网络内在特征的隐含特征，它为网络分析提供了新颖的角度<sup>[18-20]</sup>。虽然最新的一些网络嵌入方法已经在很多数据挖掘任务中取得了前景可观的效果，但是这些方法大多数只聚焦于同质网络在实际场景中，许多网络通常包含多种类型的节点和边，即异质信息网络 (HIN)<sup>[4,5,75]</sup>。以 DBLP 学术网络为例，如图 4-1(a)所示，这个网络包含四种类型的节点：作者 (Author, A)、论文 (Paper, P)、会议 (Conference, C) 和关键词 (Term, T)。网络中还包含多种类型的关系：写作/被写关系 (writing/written) 和发表/被发表关系 (publish/published) 等。此外，还有由元路径 (meta-path) 表示的一些复杂关系<sup>[36]</sup>，例如 APA (合作关系) 和 APC (作者在会议上发表论文) 等。元路径广泛用于探索捕获异质信息网络中的丰富语义信息。因此，相较于同质信息网络，异质信息网络融合了更多的信息并且包含更丰富的语义。直接将传统的同质网络嵌入方法用于异质网络嵌入，必然会降低下游数据挖掘任务的表现。

为了建模网络的异质性，有很多方法尝试嵌入异质网络。例如，一些模型利用基于元路径的随机游走生成节点序列，用于优化节点间的相似性<sup>[37,52]</sup>；一些方法将异质信息网络分解为多个简单网络，然后在每个子网络中分别优化节点间的相似性<sup>[50]</sup>；还有一些基于神经网络的方法，这些方法学习用于网络嵌入的非线性

映射函数<sup>[55]</sup>。虽然他们考虑了异质性，但是他们通常有一个假设：通过使得两个节点的表示相互靠近，单一的模型处理所有的关系和节点，如图 4-1(b)所示。

然而，异质网络中的多样关系有着明显不同的结构特征，它们需要以不同的模型区分对待。以图 4-1(a) 为例，在这个网络中有原子关系（如，AP 和 PC）和复合关系（如，APA 和 APC）。显然地，AP 关系和 PC 关系表现出相当不同的结构。也就是说，在 AP 关系中，一些作者写了一些论文，这表明了一种点对点（peer-to-peer）的对等结构。而在 PC 关系中，很多论文发表于同一个会议，这反映了一种一个点被其他点环绕（one-centered-by-another）的结构。同样地，APA 和 APC 分别表示了 peer-to-peer 和 one-centered-by-another 的结构特征。通过简单的例子，可以清晰地看到异质网络中的关系有着不同的结构特征。

对于异质网络嵌入，考虑关系的不同的结构特征是不容易的。主要有以下几个挑战：(1) 如何区分异质网络中关系的结构特征？在异质网络中，多样的关系（原子关系和元路径）包含不同的结构。(2) 如何捕获不同类别关系的不同的结构特征？由于多样的关系有不同的结构，以一个模型建模它们可能会导致信息的丢失。我们需要特别设计合适的模型，用于捕获有差异的结构特征。(3) 建模不同关系的不同模型应该能够容易地融合，并且可以联合优化求解。

为了解决以上挑战，本论文提出了一个新颖的异质信息网络嵌入模型，叫做关系结构感知的异质信息网络嵌入（RHINE）。具体地，RHINE 模型首先通过透彻的数学分析，探索了异质信息网络中关系的结构特征，并且提出了两个结构相关的度量方法。这两个方法可以将多样的关系一致地划分为两类：表示 one-centered-by-another 结构的隶属关系（Affiliation Relations, ARs）和表示 peer-to-peer 结构的交互关系（Interaction Relations, IRs）。为了捕获关系中有差异的结构特征，本论文提出了两种特别设计的模型。对于 ARs 关系，关系中的节点共享一些相似的特性<sup>[76]</sup>，通过计算节点间的欧式距离，以保证节点在隐含空间中直接相近。另一方面，对于桥接两个对等节点的 IRs 关系，将这类关系建模为节点间的平移转换。由于两个模型在数学形式上是一致的，它们可以以统一的方式联合优化求解。

综上所述，本研究模型的贡献可以总结如下：

- 首次探索了异质网络中关系的结构特征，并且提出了两个结构相关的度量方法，它们可以将异质关系一致地划分为 ARs 和 IRs。
- 提出了一个关系结构感知的异质信息网络嵌入模型（RHINE）。通过构建合适的模型，RHINE 充分建模了关系中有差异的结构特征。
- 充分的实验验证了模型 RHINE 的有效性。实验结果表明了模型 RHINE 在多个数据挖掘任务中的表现，均显著地优于现有的网络表示学习方法。

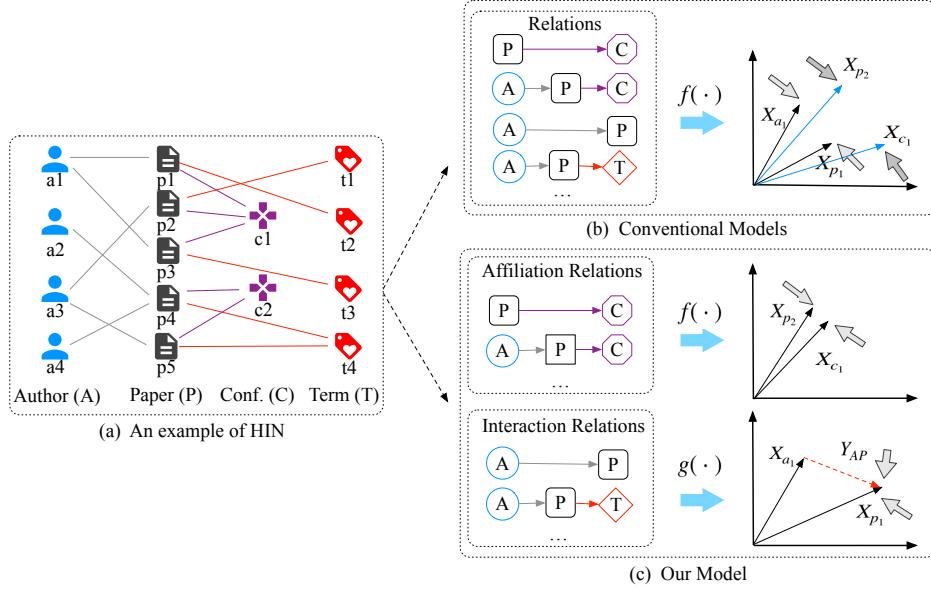


图 4-1 异质信息网络示例以及传统方法和本研究提出方法的比较。

## 4.2 异质信息网络中关系结构特性分析

### 4.2.1 数据表述

在分析关系的结构特征之前，首先简要的介绍本文所用的四个数据集，包括 DBLP、Yelp、AMiner 和 Amazon。关于数据集的详细统计可以参考表 4-1。

表 4-1 数据集统计信息

Datasets	Nodes	Number of Nodes	Relations ( $t_u \sim t_v$ )	Number of Relations	Avg. Degree of $t_u$	Avg. Degree of $t_v$	Measures	$D(r)$	$S(r)$	Relation Category
DBLP	Term (T)	8,811	PC	14,376	1.0	718.8	718.8	0.05		AR
	Paper (P)	14,376	APC	24,495	2.9	2089.7	720.6	0.085		AR
	Author (A)	14,475	AP	41,794	2.9	2.9	1.0	0.0002		IR
	Conference (C)	20	PT	88,683	6.2	10.7	1.7	0.0007		IR
Yelp	User (U)	1,286	APT	260,605	18.0	29.6	1.6	0.002		IR
	Service (S)	2	BR	2,614	1.0	1307.0	1307.0	0.5		AR
	Business (B)	2,614	BS	2,614	1.0	1307.0	1307.0	0.5		AR
	Star Level (L)	9	BL	2,614	1.0	290.4	290.4	0.1		AR
	Reservation (R)	2	UB	30,838	23.9	11.8	2.0	0.009		IR
AMiner	Paper (P)	127,623	BUB	528,332	405.3	405.3	1.0	0.07		IR
	Author (A)	164,472	PC	127,623	1.0	1263.6	1263.6	0.01		AR
	Reference (R)	147,251	APC	232,659	2.2	3515.6	1598.0	0.01		AR
	Conference (C)	101	AP	355,072	2.2	2.8	1.3	0.00002		IR
			PR	392,519	3.1	2.7	1.1	0.00002		IR
Amazon	Tag (T)	22,140	APR	1,084,287	7.1	7.9	1.1	0.00004		IR
	Item (I)	8,493	IB	8,493	1.0	386.1	386.1	0.05		AR
	User (U)	15,619	UIB	16,789	1.5	1067.9	711.9	0.05		AR
	Brand (B)	22	UI	23,493	1.5	2.7	1.8	0.0002		IR
			IT	39,528	4.6	1.8	2.6	0.0002		IR
			UIT	117,618	7.8	5.5	1.4	0.0003		IR

DBLP 是一个学术合作网络，其包含四种类型的节点：作者 (Author, A)、论文 (Paper, P)、会议 (Conference, C) 和关键词 (Term, T)。本论文基于关系集合  $\{AP, PC, PT, APC, APT\}$  抽取节点-关系元组。Yelp 是一个电商点评网络，其包含

五种类型节点：用户（User, U）、商店（Business, B）、预定类型（Reservation, R）、服务类型（Service, S）和星级（Star Level, L）。本论文考虑的关系包括： $\{BR, BS, BL, UB, BUB\}$ 。AMiner 也是一个学术网络，其包含四种类型的节点：作者（Author, A）、论文（Paper, P）、会议（Conference, C）和引用（Reference, R）。本论文考虑关系包括： $\{AP, PC, PR, APC, APR\}$ 。Amazon 是商品购买数据集，包含四种类型的节点：用户（User, U），商品（Item, I），品牌（Brand, B）和标签（Tag, T），并分析五种关系： $\{UI, IB, IT, UIB, UIT\}$ 。需要注意的是，实际上可以基于元路径分析所有的关系，但是并不是所有的元路径对网络嵌入都有正向作用<sup>[36,75]</sup>。因此，参照已有工作<sup>[37,53]</sup>，本论文选择一些重要且有意义的元路径。

#### 4.2.2 隶属关系和交互关系

##### 4.2.2.1 数据观察

由于节点的度可以很好地反映网络的结构<sup>[77]</sup>，因此首先对四个数据集上的节点度分布进行一些数据观察。具体而言，给定关系 $r$ ，在关系 $r$ 的两端都有节点类型，分别表示为 $t_u$ 和 $t_v$ 。然后，计算关系 $r$ 下的节点度的两个分布，分别表示为 $Dis(t_u|r)$ 和 $Dis(t_v|r)$ 。由于异质信息网络中的其他关系具有相似分布趋势，在这里只绘制了三个数据集上一些典型关系的度分布。对于 DBLP 数据集，在图 4-2(a) 中显示了 A 类型和 C 类型节点相对于 APC 关系的度分布，在图 4-2(d) 中展示了 A 类型和 T 类型节点相对于 APT 关系的度分布。对于 Yelp、AMiner 数据集，分别在图 4-2(b)(e) 和 (c)(f) 中绘制了节点的度分布。

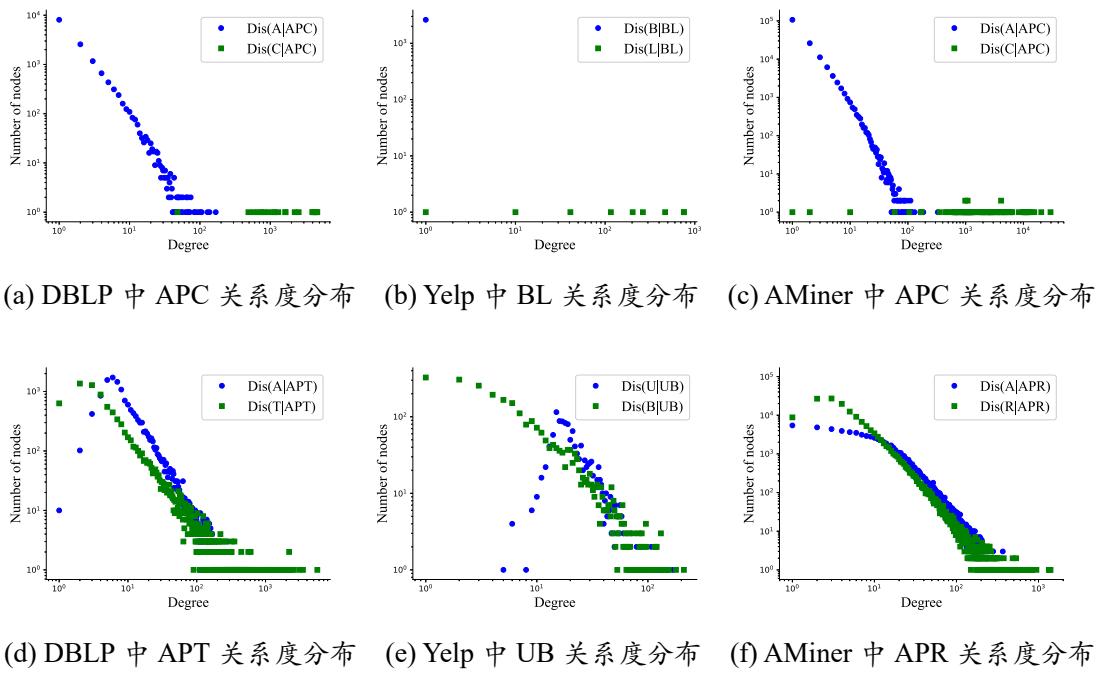


图 4-2 三个数据集中不同关系两端节点的度分布

通过观察同一数据集上不同关系连接的节点的度分布（即比较图 4-2 中的上下子图），可以发现不同的关系具有明显不同的结构特征。以 DBLP 为例，从图 4-2(a)(d)可以观察到，节点相对于 APC 和 APT 关系的度分布是完全不同的。对于 APC（即图 4-2(a)），可以发现类型为 A 的节点的度分布与类型为 C 的节点的度分布显著不同。这意味着通过关系 APC 连接的两种类型的节点是极不平衡，这种关系意味着不等价的结构。另一方面，节点相对于关系 APT 的两个度分布几乎相同，这表明通过关系 APT 连接的两种类型节点有着相似的结构角色。通过以上关于节点度分布的分析，不难发现，异质信息网络中的不同关系具有非常独特的结构特征，应仔细分析并考虑将其嵌入到异质信息网络的低维表示中。

#### 4.2.2.2 定量分析

数据观察仅能使我们直观地理解异质信息网络中关系的结构差异性，因此还需要进行合理准确的定量分析。由于节点度可以很好地反映网络的结构，本论文定义了一个基于度的度量  $D(r)$  来研究异质信息网络中关系的差异性。具体地，本论文比较由关系  $r$  连接的两种类型节点的平均度。给定关系  $r$  以及节点  $u$  和  $v$ （即，节点-关系元组  $(u, r, v)$ ）， $t_u$  和  $t_v$  分别是节点  $u$  和节点  $v$  的类型，定义  $D(r)$  如下：

$$D(r) = \frac{\max[\bar{d}_{t_u}, \bar{d}_{t_v}]}{\min[\bar{d}_{t_u}, \bar{d}_{t_v}]} \quad (4-1)$$

其中  $\bar{d}_{t_u}$  和  $\bar{d}_{t_v}$  分别是类型为  $t_u$  和  $t_v$  的节点的平均度。

$D(r)$  的数值大，表示由关系  $r$  连接的两种类型的节点之间的一种不等价结构（one-centered-by-another）。而  $D(r)$  值小，表明一种对等结构（peer-to-peer）。换言之， $D(r)$  数值大的关系表现出很强的隶属关系，由此类关系连接的节点通常共享更多相似特性<sup>[78]</sup>，而  $D(r)$  数值小的关系表示了一种交互关系。本论文将这两类关系称为隶属（Affiliation Relations, ARs）和交互关系（Interaction Relations, IRs）。

为了更好地理解多种关系间的结构差异，以 DBLP 为例说明。如表 4-1 中所示，对于 PC 关系，其  $D(PC) = 718.8$ ，类型为 P 的节点的平均度为 1.0 而类型为 C 的节点的平均度是 718.8。这表明论文和会议在结构上是不对等的，论文环绕会议。不同的， $D(AP) = 1.1$  表示作者和论文之间是一种对等（peer-to-peer）结构关系，这和常识也是一致的。在语义上，PC 关系表示“论文发表在会议上”，暗示一种隶属关系，而 AP 关系表示“作者书写论文”，明显描述了一种交互关系。

事实上，本论文还定义了一些其他的度量方法来捕获关系结构的差异性。例如，就稀疏性比较关系，其可以被定义为：

$$S(r) = \frac{N_r}{N_{t_u} \times N_{t_v}} \quad (4-2)$$

其中  $N_r$  表示关系实例的数量， $N_{t_u}$  和  $N_{t_v}$  分别是类型为  $t_u$  和  $t_v$  的节点的数量。这个度量方法可以一致地将关系划分为两类：ARs 和 IRs。

显然地，隶属关系（ARs）和交互关系（IRs）表现出相当不同的特征：（1）ARs 表示了 one-centered-by-anther 的结构，关系中的两类节点的平均度差异非常大。（2）IRs 描述了 peer-to-peer 的结构，关系中的两类节点的平均度是对等的。

## 4.3 模型设计与实现

### 4.3.1 符号及定义

#### 定义 4.1：异质信息网络

一个异质信息网络通常被定义为一个图  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{O}, \mathcal{R})$ ，其中  $\mathcal{V}$  和  $\mathcal{E}$  分别表示节点和边关系的集合。每个节点  $v \in \mathcal{V}$  和边  $e \in \mathcal{E}$  分别关联一个类型映射函数  $\phi: \mathcal{V} \rightarrow \mathcal{O}$  和  $\varphi: \mathcal{E} \rightarrow \mathcal{R}$ ，其中  $\mathcal{O}$  和  $\mathcal{R}$  表示节点和边的类型集合， $|\mathcal{O}| + |\mathcal{R}| > 2$ 。

#### 定义 2：元路径

对于异质信息网络  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{O}, \mathcal{R})$ ，长度为  $l$  的元路径可以定义为一种复合关系： $\mathcal{P} = o_1 \xrightarrow{r_1} o_2 \xrightarrow{r_2} \dots \xrightarrow{r_l} o_{l+1}$ ，其中  $o_i \in \mathcal{O}$  且  $r_i \in \mathcal{R}$ 。如果两个对象/节点之间有且只有一种关系类型，上述元路径可以简写为  $\mathcal{P} = o_1 o_2 \dots o_{l+1}$ 。

#### 定义 3：节点-关系元组

在一个异质信息网络中，关系  $R$  包括原子关系（例如，链接）和复合关系（例如，元路径）。一个节点-关系元组  $\langle u, r, v \rangle \in P$  描述了两个节点  $u$  和  $v$  由关系  $r \in R$  连接。这里的  $P$  表示节点-关系元组集合。如图 4-1(a) 所示， $\langle a_2, APC, c_2 \rangle$  是一个节点-关系元组，表示  $a_2$  写了一篇论文并发表在  $c_2$  上。

#### 定义 4：异质信息网络表示学习

给定一个异质信息网络  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{O}, \mathcal{R})$ ，异质信息网络表示学习的目标是学习一个映射函数  $f: \mathcal{V} \rightarrow \mathbb{R}^d$ ，其可以将每个节点  $v \in V$  映射到一个低维的向量空间  $\mathbb{R}^d$ ，其中  $d \ll |V|$  且低维向量空间保留异质信息网络的复杂结构和丰富语义。

### 4.3.2 基本思路

通过透彻的数学分析，发现可以依据关系差异化的结构特征，明显地将异质关系区分为 ARs 和 IRs。为了充分建模关系的差异化特征，需要为不同类别的关系特别地设计不同且合适的模型。对于 ARs，本论文提出以欧式距离作为度量标准来衡量节点间的相似性，其背后的动机是：（1）首先，ARs 表现了节点之间的隶属结构，其表明由此类关系连接的节点共享相似的特性<sup>[76,78]</sup>。因此，在表示向量空间中，由 ARs 连接的节点可以直接相互靠近，这和欧式距离的优化目标也是一

致的<sup>[79]</sup>。(2) 此外, 异质信息网络嵌入的一个目标是保留网络中的高阶相似性。由于欧式距离满足三角不等关系<sup>[80]</sup>, 其可以保留网络的一阶和二阶相似性。

不同于 ARs, IRs 表明了节点之间的交互关系, 关系本身包含了节点间结构信息。本论文提出显式地将 IRs 建模为节点间的平移操作。此外, 基于平移的距离与欧式距离在数学形式上是一致的<sup>[81]</sup>, 因此可以容易地结合并联合优化求解。

### 4.3.3 建模 ARs 和 IRs 的不同模型

#### 4.3.3.1 欧式距离建模隶属关系

由 ARs 连接的节点共享相似的特性<sup>[78]</sup>, 在表示向量空间中可以直接相互靠近。本论文以欧式距离度量节点之间的相似性。给定隶属节点-关系元组  $\langle p, s, q \rangle \in P_{AR}$ , 其中  $s \in R_{AR}$  是  $p$  和  $q$  之间的带有权重值  $w_{pq}$  的关系,  $p$  和  $q$  间距离定义为:

$$f(p, q) = w_{pq} \|\mathbf{X}_p - \mathbf{X}_q\|_2^2 \quad (4-3)$$

其中  $\mathbf{X}_p \in \mathbb{R}^d$  和  $\mathbf{X}_q \in \mathbb{R}^d$  分别是节点  $p$  和  $q$  的嵌入向量。由于  $f(p, q)$  度量了节点  $p$  和  $q$  在低维表示空间的距离, 我们旨在最小化  $f(p, q)$  来确保由 AR 连接的节点应该相互靠近。因此, 我们定义如下基于间隔的损失函数<sup>[81]</sup>:

$$\mathcal{L}_{EuAR} = \sum_{s \in R_{AR}} \sum_{\langle p, s, q \rangle \in P_{AR}} \sum_{\langle p', s, q' \rangle \in P'_{AR}} \max[0, \gamma + f(p, q) - f(p', q')] \quad (4-4)$$

其中  $\gamma > 0$  是间隔超参数,  $P_{AR}$  是正样本, 而  $P'_{AR}$  是负样本。

#### 4.3.3.2 基于平移的距离建模交互关系

交互关系表明了对等节点之间很强的交互结构, 因此不同于 ARs, 可以显式地建模 IRs 为节点间的平移操作。形式化地, 给定一个交互节点-关系元组  $\langle u, r, v \rangle$ , 其中  $r \in R_{IR}$  是带有权重  $w_{uv}$  的关系, 我们定义如下的函数:

$$g(u, v) = w_{uv} \|\mathbf{X}_u + \mathbf{Y}_r - \mathbf{X}_v\| \quad (4-5)$$

其中  $\mathbf{X}_u$  和  $\mathbf{X}_v$  是节点  $u$  和  $v$  的嵌入向量,  $\mathbf{Y}_r$  是关系  $r$  的嵌入向量。显然, 上述函数惩罚  $(\mathbf{X}_u + \mathbf{Y}_r)$  与  $\mathbf{X}_v$  的偏差。对于交互节点-关系元组  $\langle u, r, v \rangle \in P_{IR}$ , 定义损失函数:

$$\mathcal{L}_{TrIR} = \sum_{r \in R_{IR}} \sum_{\langle u, r, v \rangle \in P_{IR}} \sum_{\langle u', r, v' \rangle \in P'_{IR}} \max[0, \gamma + f(u, v) - f(u', v')] \quad (4-6)$$

其中  $P_{IR}$  是正的交互节点-关系元组, 而  $P'_{IR}$  是负交互节点-关系元组。

#### 4.3.3.3 同一模型嵌入异质信息网络

最后, 融合建模不同关系的两个子模型, 并最小化如下损失函数:

$$\begin{aligned} \mathcal{L} &= \mathcal{L}_{EuAR} + \mathcal{L}_{TrIR} \\ &= \sum_{s \in R_{AR}} \sum_{\langle p, s, q \rangle \in P_{AR}} \sum_{\langle p', s, q' \rangle \in P'_{AR}} \max[0, \gamma + f(p, q) - f(p', q')] \\ &\quad + \sum_{r \in R_{IR}} \sum_{\langle u, r, v \rangle \in P_{IR}} \sum_{\langle u', r, v' \rangle \in P'_{IR}} \max[0, \gamma + f(u, v) - f(u', v')] \end{aligned} \quad (4-7)$$

#### 4.3.3.4 采样及优化

如表 4-1 所示, ARs 和 IRs 的分布是不平衡的, 而且两类关系内部的关系分布也是不平衡的。传统的边采样可能会导致数量较少的边过采样, 而数量较多的边欠采样。为了解决这一问题, 本论文根据关系概率分布采样正样本。对于负样本, 参照已有工作<sup>[81,82]</sup>构建负节点-关系元组集合  $P'_{u,r,v} = \{(u', r, v) | u' \in V\} \cup \{(u, r, v') | v' \in V\}$ , 其中要么随机替换头节点要么替换尾节点, 但不会同时被替换。

对于目标函数, 本论文采用随机梯度下降算法进行优化。算法逐步优化基于欧式距离的模型和基于平移的距离模型。具体来说, 首先初始化具有均匀分布的节点嵌入矩阵  $\mathbf{X}$  和关系嵌入矩阵  $\mathbf{Y}$ 。然后, 对正和负交互节点关系三元组进行采样, 并相应地更新  $\mathbf{X}$  和  $\mathbf{Y}$ 。以同样的方式, 对正和负隶属节点关系三元组进行采样, 并且仅更新  $\mathbf{X}$ 。最后, 返回节点嵌入矩阵  $\mathbf{X}$  和关系嵌入矩阵  $\mathbf{Y}$ 。

### 4.4 实验及分析

#### 4.4.1 实验设置

##### 4.4.1.1 数据集

如 4.2.1 所述, 本章节在四个数据集上进行了实验, 包括 DBLP, Yelp, AMiner 和 Amazon。表 4-1 汇总了四个数据集的统计信息。

- DBLP 是学术合作数据集, 由 14456 位作者, 14375 篇论文, 8811 个术语和 20 个会议组成。有 200 篇论文标有其研究领域作为标签。
- Yelp 是 Yelp Challenge 提供的社交媒体数据集。本实验抽取与以下三个子类别的商店有关的信息: “美国食品”, “快餐” 和 “寿司”, 并构建了一个异质信息网络, 其中包括 1286 个用户, 2614 个商店, 9 个星级级别, 2 种服务类型和 2 种预订类型。本实验对商店类型节点进行聚类和分类。
- AMiner 是学术合作网络, 包括 164472 位作者, 127623 篇论文, 147251 篇参考文献和 101 个会议。本实验以与计算机科学相关的 10 个领域作为论文类型节点的标签, 并对论文类型节点进行聚类和分类。
- Amazon 是产品购买数据集。本实验抽取信息并构建一个异质信息网络, 其中包含 15619 个用户, 8493 个商品, 22140 个标签和 22 个品牌。本实验将 9 个类别作为商品的标签, 并对商品类型节点进行聚类和分类。

##### 4.4.1.2 对比方法

该实验将本论文提出的 RHINE 模型与 7 种网络表示学习方法进行了比较, 其中 DeepWalk<sup>[24]</sup> 和 LINE<sup>[46]</sup> 是针对同质网络而设计的, 其余的则能够建模异质信息网络, 即 PTE<sup>[50]</sup>、ESim<sup>[53]</sup>、HIN2vec<sup>[52]</sup>、Metapath2vec<sup>[37]</sup>、HERec<sup>[38]</sup> 和 JUST<sup>[83]</sup>。

#### 4.4.1.3 参数设置

该实验设置所有模型的网络表示维度  $d=100$ , 负样本数量  $k=3$ 。对于 DeepWalk, HIN2Vec, metapath 和 JUST, 每个节点的游走次数为 10, 游走长度为 100, 窗口大小为 5。对于 RHINE, 损失函数中的间隔参数  $\gamma$  设置为 1, 学习率设置为 0.005, 训练批次大小和迭代次数分别设置为 128 和 400。

#### 4.4.2 节点聚类实验

本小节进行节点聚类实验, 以说明通过网络表示学习到的节点嵌入表示如何有益于异质信息网络中的节点聚类任务。在该任务中, 利用上述网络表示学习方法学习节点的低维向量表示。基于学习到的节点表示, 利用 K-means 算法对节点进行聚类, 并根据归一化互信息 (NMI) 指标评估聚类结果。

表 4-2 节点聚类实验结果

Methods	DBLP	Yelp	AMiner	Amazon
DeepWalk	0.3884	0.3043	0.5427	0.7062
LINE-1st	0.2775	0.3103	0.3736	0.5997
LINE-2nd	0.4675	0.3593	0.3862	0.6709
PTE	0.3101	0.3527	0.4089	0.6692
ESim	0.3449	0.2214	0.3409	0.6403
HIN2Vec	0.4256	0.3657	0.3948	0.6794
metapath2vec	0.6065	0.3507	0.5586	0.7145
HERec	0.5893	0.3313	0.5123	0.6878
JUST	0.5634	0.3516	0.5021	0.6894
RHINE	0.7204	0.3882	0.6024	0.7435
RHINE-M	<b>0.7323</b>	<b>0.3934</b>	<b>0.6152</b>	<b>0.7582</b>

如表 4-2 所示, 本文提出的模型 RHINE 明显优于所有比较方法。(1) 与最佳对比方法相比, 模型 RHINE 的聚类性能在 DBLP, Yelp, AMiner 和 Amazon 上分别提高了 18.79%, 6.15%, 7.84%, 6.12%。显著的效果提升表明了 RHINE 模型的有效性, 其通过区分异质信息网络中具有不同结构特征的各种关系来更好地嵌入异质信息网络。此外, 实验结果还验证了针对不同类别关系设计的模型的有效性。(2) 在所有对比方法中, 同质信息网络表示学习模型的表现效果都欠佳, 这是因为这类方法忽略了网络中节点和边连接的类型, 从而丢失了一些信息。(3) 在所有数据集上, RHINE 明显优于现有的异质信息网络表示学习模型(即 ESim, HIN2Vec, metapath2vec, HERec 和 JUST), 原因是 RHINE 模型针对不同关系类别提出的具有适当模型的方法可以更好地捕获异质信息网络的结构和语义信息。

#### 4.4.3 链路预测实验

链路预测旨在预测网络中两个节点之间存在链路的可能性。形式上, 给定节点对  $(u, v)$ , 该任务旨在预测网络中它们之间是否存在关系  $r$ 。在该实验中,

将链接预测问题建模为旨在预测链接是否存在的二分类问题。对于 DBLP 和 AMiner 数据集，进行合作关系预测（即 A-P-A）和作者参会关系预测（即 A-P-C）。对于 Yelp 和 Amazon 数据集，进行消费关系预测（即 U-B 和 U-I）。在实验中，首先将原始异质信息网络随机分为训练网络和测试网络，其中训练网络包含 80% 的预测关系（即 A-A, A-C, U-B 和 U-I），而测试网络包含其余关系。然后，在训练网络上训练模型学习节点的嵌入向量，并在测试网络上评估预测性能。

表 4-3 中报告了有关 AUC 和 F1 分数的链接预测任务的结果。显然，RHINE 模型在四个数据集上的性能均优于所有对比方法。链路预测结果提升的原因在于，RHINE 基于欧几里得距离建模关系的模型可以捕获一阶和二阶相似性。此外，RHINE 根据其结构特征将多种类型的关系分为两类，从而可以更好地学习节点的嵌入表示，这对于预测两个节点之间的复杂关系是有利的。

表 4-3 链路预测实验结果

Methods	DBLP (A-A)				DBLP (A-C)				Yelp (U-B)				AMiner (A-A)				AMiner (A-C)				Amazon (U-I)			
	AUC	F1	AUC	F1	AUC	F1	AUC	F1	AUC	F1														
DeepWalk	0.9131	0.8246	0.7634	0.7047	0.8476	0.6397	0.9122	0.8471	0.7701	0.7112	0.9045	0.8978												
LINE-1st	0.8264	0.7233	0.5335	0.6436	0.5084	0.4379	0.6665	0.6274	0.7574	0.6983	0.8810	0.8619												
LINE-2nd	0.7448	0.6741	0.8340	0.7396	0.7509	0.6809	0.5808	0.4682	0.7899	0.7177	0.8879	0.8793												
PTE	0.8853	0.8331	0.8843	0.7720	0.8061	0.7043	0.8119	0.7319	0.8442	0.7587	0.9014	0.8991												
ESim	0.9077	0.8129	0.7736	0.6795	0.6160	0.4051	0.8970	0.8245	0.8089	0.7392	0.8992	0.8875												
HIN2Vec	0.9160	0.8475	0.8966	0.7892	0.8653	0.7709	0.9141	0.8566	0.8099	0.7282	0.9012	0.8981												
metapath2vec	0.9153	0.8431	0.8987	0.8012	0.7818	0.5391	0.9111	0.8530	0.8902	0.8125	0.9388	0.9145												
HERec	0.9178	0.8523	0.8874	0.8132	0.7923	0.5742	0.9058	0.8514	0.8834	0.8067	0.9295	0.9017												
JUST	0.9018	0.8103	0.8174	0.7532	0.7711	0.5932	0.8823	0.8141	0.8045	0.7603	0.8955	0.8912												
RHINE	0.9315	0.8664	0.9148	0.8478	0.8762	0.7912	0.9316	0.8664	0.9173	0.8262	0.9561	0.9207												
RHINE-M	<b>0.9421</b>	<b>0.8726</b>	<b>0.9218</b>	<b>0.8508</b>	<b>0.8796</b>	<b>0.7994</b>	<b>0.9460</b>	<b>0.8714</b>	<b>0.9207</b>	<b>0.8322</b>	<b>0.9634</b>	<b>0.9371</b>												

#### 4.4.4 节点分类实验

多类分类是评估网络表示学习性能的一项常见任务。具体地，在学习了节点表示向量之后，利用 40%, 60% 和 80% 的标记节点训练了一个逻辑回归分类器，并用剩余的数据进行测试。分类结果以 Micro-F1 和 Macro-F1 作为评估指标。

实验结果如表 4-4 所示，正如我们可以观察到的，(1) 在除 AMiner 之外的所有数据集上，RHINE 的分类性能均优于所有对比方法。平均地，模型在 DBLP 和 Yelp 数据集上的节点分类性能提高了约 4%。在 Amazon 数据集上，模型 RHINE 在大多数情况下仍然表现良好。在 AMiner 方面，RHINE 比 ESim, HIN2Vec 和 metapath2vec 表现稍差。这可能是由于过度捕获关系 PR 和 APR 的信息引起的（R 表示引用类型节点）。由于作者可能会写一篇涉及各种领域的论文，因此这些关系可能会带来一些噪音。(2) 尽管 ESim, HIN2Vec 和 JUST 可以对异质信息网络中的多种类型的关系进行建模，但它们在大多数情况下均无法很好地发挥作用。本论文提出的模型 RHINE 由于建模了各种关系的独特特性而取得了良好

的性能。(3)本文的 RHINE 模型在不同训练比率下的稳定性能表明,模型学习到的节点低维嵌入表示对维度大小的鲁棒性。

表 4-4 节点分类实验结果

Datasets	Metrics	Tr.Ra	D.W.	L-1st	L-2nd	PTE	ESim	H2Vec	mp2vec	HERec	JUST	RHINE	RHINE-M
DBLP	Ma.-F1	40%	0.6754	0.7237	0.7488	0.7723	0.8069	0.9073	0.9084	0.9005	0.8612	0.9261	<b>0.9301</b>
		60%	0.7283	0.7311	0.7338	0.7768	0.8161	0.8600	0.8941	0.8963	0.8733	0.9257	<b>0.9387</b>
		80%	0.7475	0.8091	0.7559	0.8852	0.8867	0.8631	0.8976	0.9101	0.8745	0.9344	<b>0.9445</b>
	Mi.-F1	40%	0.6937	0.7250	0.7583	0.7629	0.8000	0.9083	0.9098	0.9000	0.8567	0.9125	<b>0.9237</b>
		60%	0.7125	0.7500	0.7375	0.7875	0.8125	0.8625	0.8950	0.8915	0.8654	0.9051	<b>0.9254</b>
		80%	0.7500	0.8250	0.7500	0.8750	0.8750	0.8500	0.9000	0.9021	0.8705	0.9250	<b>0.9304</b>
Yelp	Ma.-F1	40%	0.6708	0.4607	0.5393	0.5397	0.6799	0.6104	0.5613	0.5509	0.5583	0.6909	<b>0.6998</b>
		60%	0.6717	0.4681	0.5299	0.5407	0.6830	0.6032	0.5478	0.5557	0.5632	0.7021	<b>0.7110</b>
		80%	0.6723	0.4872	0.5304	0.5389	0.6836	0.6075	0.5337	0.5517	0.5701	0.7132	<b>0.7205</b>
	Mi.-F1	40%	0.6732	0.6680	0.6637	0.7297	0.7247	0.7342	0.7074	0.7265	0.6002	0.7476	<b>0.7523</b>
		60%	0.6893	0.6537	0.6857	0.7323	0.7358	0.7189	0.7171	0.7304	0.6201	0.7562	<b>0.7598</b>
		80%	0.7012	0.6639	0.7377	0.7342	0.7399	0.7361	0.7208	0.7323	0.6295	0.7572	<b>0.7634</b>
AMiner	Ma.-F1	40%	0.9421	0.9473	0.9392	0.9649	0.9898	<b>0.9955</b>	0.9895	0.9901	0.9535	0.9798	0.9832
		60%	0.9423	0.9456	0.9467	0.9736	0.9904	<b>0.9961</b>	0.9912	0.9942	0.9684	0.9845	0.9893
		80%	0.9386	0.9494	0.9468	0.9791	0.9910	<b>0.9962</b>	0.9934	0.9956	0.9745	0.9884	0.9923
	Mi.-F1	40%	0.9525	0.9471	0.9397	0.9754	0.9936	<b>0.9958</b>	0.9901	0.9874	0.9684	0.9804	0.9848
		60%	0.9533	0.9527	0.9484	0.9813	0.9943	<b>0.9962</b>	0.9916	0.9953	0.9734	0.9845	0.9897
		80%	0.9512	0.9569	0.9491	0.9874	0.9948	<b>0.9965</b>	0.9936	0.9948	0.9844	0.9807	0.9912
Amazon	Ma.-F1	40%	0.9618	0.9596	0.9636	0.9657	0.9713	0.9875	0.9814	0.9822	0.9732	0.9958	<b>0.9969</b>
		60%	0.9645	0.9601	0.9623	0.9703	0.9759	0.9896	0.9871	0.9814	0.9766	0.9963	<b>0.9972</b>
		80%	0.9703	0.9652	0.9679	0.9721	0.9832	<b>0.9982</b>	0.9898	0.9853	0.9801	0.9967	0.9978
	Mi.-F1	40%	0.9634	0.9651	0.9658	0.9634	0.9689	0.9759	0.9866	0.9845	0.9699	0.9864	<b>0.9870</b>
		60%	0.9691	0.9672	0.9649	0.9678	0.9702	0.9819	0.9856	0.9878	0.9732	0.9961	<b>0.9892</b>
		80%	0.9734	0.9721	0.9714	0.9701	0.9734	<b>0.9974</b>	0.9872	0.9882	0.9764	0.9931	0.9939

#### 4.4.5 节点推荐实验

与链接预测不同,节点推荐实验可被视为排序任务。此外,节点推荐从节点视图评估网络表示学习方法的性能,这与链路预测从边关系的角度来评估性能是不同的。在此任务中,旨在向 DBLP 和 AMiner 网络中的作者推荐会议(即 A-C 关系),为 Yelp 和 Amazon 网络中的用户推荐商品(即 U-B 和 U-I)。具体来说,首先将原始数据集分为训练集和测试集,然后在训练集上学习节点嵌入表示。然后,对于测试集中的每个节点  $v_j$ ,计算节点  $v_i$  的嵌入表示与网络中其他节点的嵌入表示之间的相似度,称为两个节点之间的排名得分。之后,对节点  $v_i$  与其他节点的排名得分进行排序,并得出得分最高的前  $k$  个节点作为候选者。最后,将真实候选节点与预测结果进行比较,并使用 Hit@k 评估推荐的质量。

图 4-3 节点推荐实验结果报告了四个数据集上节点推荐的结果。可以发现,就 Hit@k 而言,本论文提出的 RHINE 的性能持续优于所有对比方法。可以认为,显著的提升是由于模型 RHINE 保留了异质信息网络中的高阶相似性。此外,所提出的 RHINE 考虑到了独特的结构特征和语义信息,从而有效地保证了通过异质关系连接的各种类型节点的嵌入向量精度。

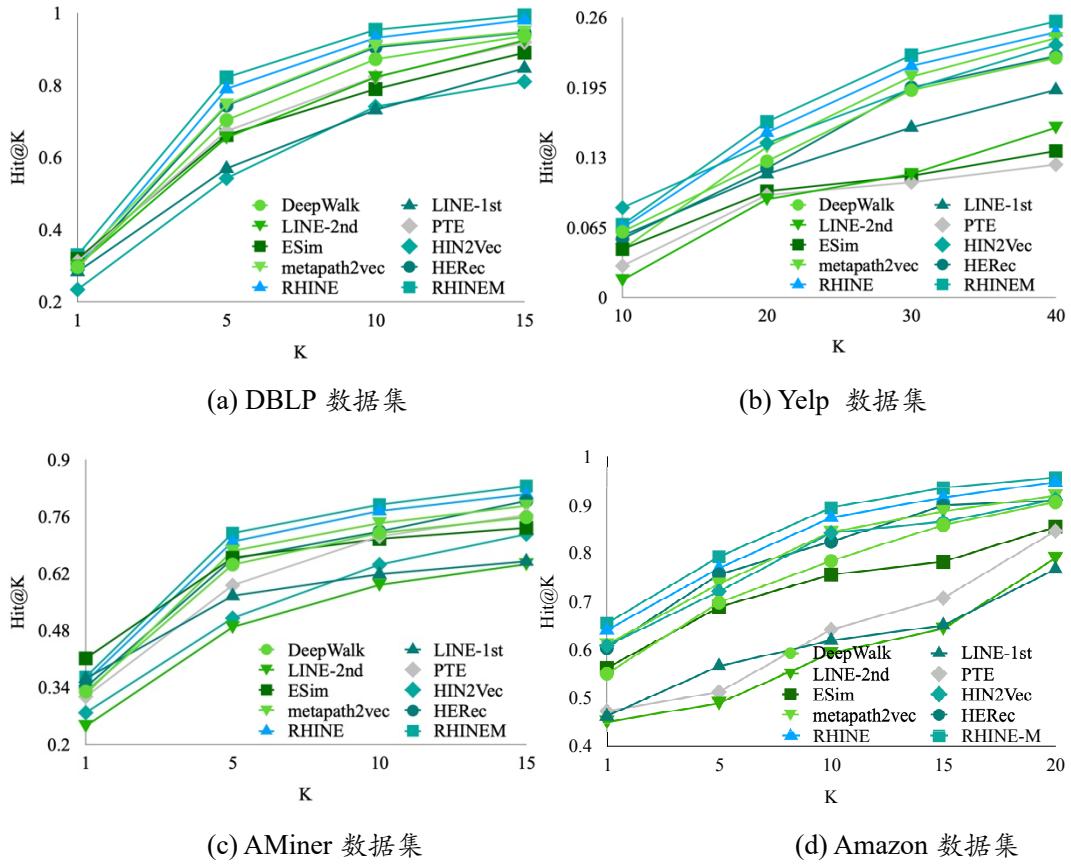


图 4-3 节点推荐实验结果

#### 4.4.6 模型分析实验

##### 4.4.6.1 模型设计分析实验

为了验证区分关系结构特征的有效性，基于 RHINE 设计了三个变体模型：

- RHINE<sub>Eu</sub> 不区分异质信息网络中的多种关系，而仅利用欧式距离来嵌入网络。
- RHINE<sub>Tr</sub> 用基于平移的距离来建模异质信息网络中的所有节点和关系，而与关系的不同类别无关，就像 TransE<sup>[81]</sup>。
- RHINE<sub>Re</sub> 利用欧几里得距离建模交互关系，基于平移的距离来建立隶属关系。

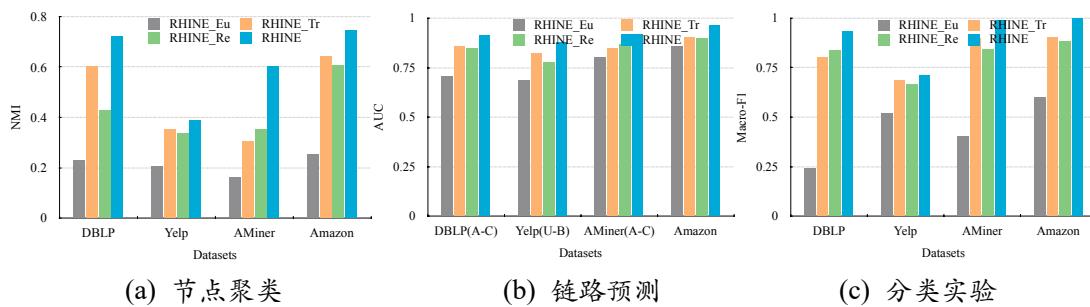


图 4-4 模型分析实验

在该实验中，变体模型的参数与 RHINE 的参数设置相同。在节点聚类、链路预测和节点分类任务上，实验结果如图 4-4 所示。很明显，完整模型 RHINE 优于 RHINE<sub>Eu</sub> 和 RHINE<sub>Tr</sub>，这表明通过区分异质关系来学习节点嵌入表示是有益的。

此外，可以发现 RHINE<sub>Tr</sub> 比 RHINE<sub>Eu</sub> 具有更好的性能，这是由于网络中通常存在更多的对等关系（即，IR），直接使所有节点彼此靠近会导致大量信息丢失。与反向模型 RHINE<sub>Re</sub> 相比，RHINE 在所有任务上也具有更好的性能，意味着 AR 和 IR 的两个模型经过精心设计，可以捕捉到它们的独特特征。

#### 4.4.6.2 可视化分析实验

为了直观地理解网络低维嵌入的表示形式，该实验在 DBLP 中可视化了节点（例如论文类型）的嵌入向量。具体来说，利用 t-SNE<sup>[84]</sup> 在二维空间中投射向量，并在图 4-5 中报告了 DeepWalk, metapath2vec 和 RHINE 的结果。

显然地，RHINE 可以清晰地将具有明显边界的不同数据集中的节点分开。如我们所见，潜在空间中的论文类型节点被分为四类，这与论文节点的四个领域是一致的。此外，每组彼此之间都很好地分开，这表明 RHINE 通过区分网络中的异构关系能够学习高级节点嵌入向量。相反，DeepWalk 和 LINE 几乎没有将节点分成不同的组。由于 ESim, HIN2Vec 和 Metapath2vec 是为异质信息网络设计的，因此它们的性能优于 DeepWalk 和 LINE，但各组之间的边界较为模糊。

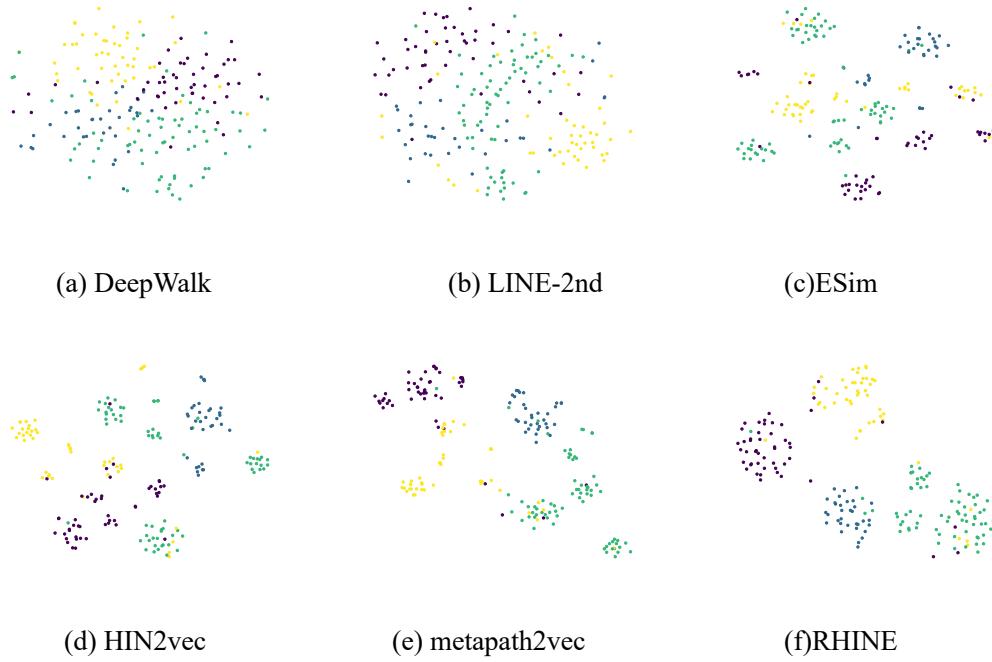


图 4-5 可视化实验

#### 4.4.6.3 参数分析实验

为了评估所提模型 RHINE 中不同参数对实验结果的影响，该实验研究了参数对节点聚类任务的敏感性（其他任务上有相似的表现）。具体来说，该实验研究了两个参数的敏感性，包括负样本数和节点表示向量的维数。在该实验中，嵌入维数的数量设置为 10、50、100、200 和 300，负样本的数量设置为 1、2、3、5 和 7。节点聚类的结果如图 4-6 所示。

如图 4-6(a)所示,模型 RHINE 的性能随着节点表示向量维数的增加而提高,然后在维数达到 100 左右时趋于稳定。显然, RHINE 能够使用低维度的向量表示捕获异质信息网络中各种关系的丰富信息。同样,图 4-6(b)显示,随着负样本数量的增加,模型的性能首先增长,然后在数量达到 3 时变得稳定。总体而言,变化趋势是平稳的,表明所提出的 RHINE 对这两个参数不是非常敏感。

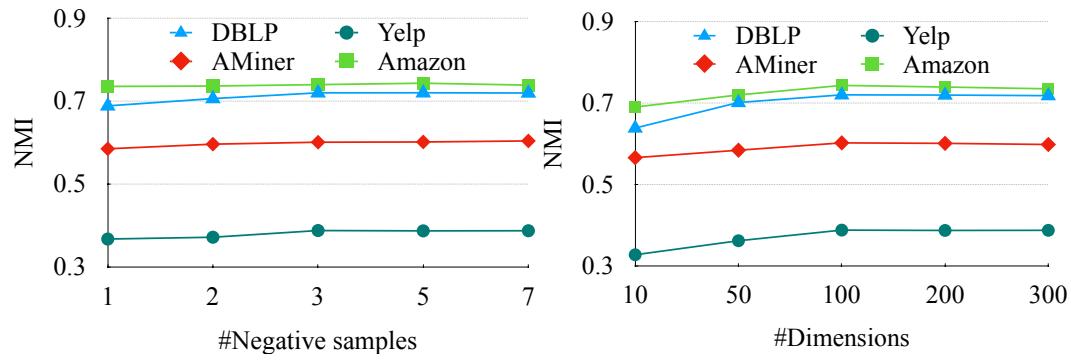


图 4-6 参数实验

## 4.5 本章小结

本章内容介绍了关系结构感知的异质信息网络表示学习算法 (RHINE)。在 4.1 节,介绍了模型 RHINE 的提出背景及创新贡献,并概述了模型的基本思想。4.2 节对异质信息网络中的关系结构特性进行了透彻的探究,以更好的理解模型 RHINE 的研究动机。其中 4.2.1 小节介绍了本研究所使用了四个数据集,4.2.2 小节定义了异质信息网络中的两类关系。在 4.3 节,详细地叙述了模型的设计与实现,首先 4.3.1 小节对相关符号和定义进行了介绍,其次 4.3.2 小节概述了模型的基本思想,然后 4.3.3 小节详细介绍了对异质信息网络中两类关系的建模方法。在 4.4 节,主要报告了针对模型 RHINE 的实验和分析,包括 4.4.1 小节的实验设置,4.4.2 小节到 4.4.5 小节对模型有效性的验证实验,4.4.6 小节对模型的分析,以及 4.4.6.2 小节对网络嵌入表示的可视化和对模型参数的分析。

## 第五章 动态异质信息网络表示学习算法

异质信息网络表示学习旨在学习节点的低维向量表示，同时保留异质信息网络中的结构和语义信息。已有的异质信息网络表示方法主要聚焦于静态网络。但是，在实际中，异质信息网络通常随着时间的推进而发生演变，这种演变包括多种类型的节点和边的增加（删除）。即使一个微小的变化也会影响整个异质信息网络结构和语义信息的变化，因此网络的动态性给异质信息网络的表示学习带来了巨大挑战。为了捕获网络的演变，传统的异质信息网络表示学习方法需要在每个时间步上进行重新训练，这是非常耗时且不切实际的。在本论文中，研究了动态异质信息网络表示学习的问题，并提出了一个新颖的异质信息网络表示学习模型（DyHNE）。更具体地说，本论文首先设计了一个基础模型 StHNE，用于保留网络中的结构和语义信息。基于谱理论，该模型通过解决广泛特征值问题，学习节点的表示向量。随着异质信息网络的演化，利用元路径增强的邻接矩阵的摄动，捕获网络的变化。然后基于特征值摄动理论，模型可以高效地更新节点的表示向量。在三个真实的异质信息网络上进行了丰富的实验，实验结构表明所提模型在有效性和效率性上均优于最新的表示学习方法。

### 5.1 研究背景

在真实世界中，异质信息网络通常表现出很强的动态性，即各种类型节点和边的增加（删除）<sup>[9,11]</sup>。此外，在异质信息网络中，节点和边的变化会因为类型的不同而有所区别。以学术网络为例，在不同的年份，导师可能会和不同的学生合作不同的论文，这会导致合作关系和论文节点的持续演变。除此以外，每年有大量新的论文节点被加入到网络中，而会议节点的数量基本保持不变。考虑到动态异质信息网络中的复杂演变，已有的表示学习模型几乎不能处理这种动态性。在异质信息网络中，即使网络发生很小的变化，这些方法需要在每个时间步上重新训练，这种做法时非常耗时并且不合实际的。尽管有研究者针对动态网络提出了一些网络表示学习模型<sup>[32,34,65]</sup>，但是他们没有考虑网络的异质性并且忽略了异质信息网络中多样的语义关系。直接利用这些方法建模动态异质信息网络，必然会丢失一些接否和语义信息，这会导致下游任务表现不佳。因此，在实际异质信息网络分析中，非常需要一种有效且高效的动态异质信息网络表示学习方法。

从根本上来说，在动态异质信息网络表示学习中，有两个基本问题需要被仔细考虑。第一个问题，如何有效地在动态环境中保留网络的拓扑结构和语义关系。

由于拓扑结构和语义关系是异质信息网络中两个最重要且直接的信息，它们从本质上保证了学到的节点表示的有效性。当有一个新的节点加入到异质信息网络中，以节点为中心的局部结构已发生变化。同时，这种变化会通过不同的元路径逐渐传递到所有的节点，这会导致整体结构的变化。此外，新节点不仅会和邻居建立直接连接，同时也会通过各种元路径和其他节点建立复杂的关系，这必然会影响异质信息网络中复杂的语义关系。因此，随着动态异质信息网络的演变，网络的拓扑结构和语义关系都会发生改变。对于一个有效的动态异质信息网络表示学习模型，同时建模这种变化并编码结构和语义信息是很有必要但也是充满挑战的。

第二个问题，当网络发生演变时，如何高效地更新节点的表示而不重新训练更新异质信息网络。对于每个时间步，重新训练一个异质信息网络表示学习模型是获得最优节点表示的最直接方法。然而，显然地，这种策略是非常耗时的，尤其当网络的结构发生很微小变化的时候。在大数据时代，重新训练的方式是不实际的。因此，构建一个切实可行的动态异质信息网络表示学习方法，以此更新节点的表示是非常有必要的。

本文提出了一个动态异质信息网络表示学习模型(DyHNE)来有效且高效地学习节点的表示。受广泛用于捕获系统变化的扰动理论<sup>[85]</sup>的启发，可以通过求解广义特征值问题来学习节点嵌入向量表示，并利用特征值扰动对异质信息网络的演化进行建模。沿着这个研究路线，该课题首先采用元路径增强的邻接矩阵来建模异质信息网络的拓扑结构，然后构建一个基础静态异质信息网络表示学习模型，即StHNE，来保留基于元路径的一阶相似性和二阶相似性。因此，静态模型StHNE可以更好地捕获动态异质信息网络中的结构和语义信息。为了捕获异质信息网络的演变，利用多个元路径增强邻接矩阵的扰动以自然的方式建模异质信息网络的结构和语义的变化。更进一步地，基于特征值摄动理论捕获网络的变化并高效地求解网络节点的表示。

综上所述，本研究算法主要贡献可以总结为如下几点：

- 该研究是第一个致力于动态异质信息网络表示学习问题的工作，其能够捕捉网络中动态演变的本质，并在现实应用中变得更加实用。
- 提出了一个动态异质信息网络表示学习模型 DyHNE，该模型能够捕获动态异质信息网络中的结构和语义信息。此外，DyHNE 模型可以有效且高效地学习节点的表示向量，模型的时间复杂度与节点的数量呈线性关系，这可以保证 DyHNE 模型可以用于大规模异质信息网络。
- 在三个数据集上进行了全面的实验。在各类任务中，实验结果表明所提模型在有效性和效率方面明显优于最新的表示学习模型。

## 5.2 算法设计及实现

### 5.2.1 符号及定义

#### 定义 1：动态异质信息网络

在 $t$ 时间步上的动态异质信息网络定义为 $\mathcal{G}^t = (\mathcal{V}^t, \mathcal{E}^t, \mathcal{O}, \mathcal{R})$  ( $1 \leq t \leq T$ )，其中 $\mathcal{V}^t$ 和 $\mathcal{E}^t$ 分别表示在 $t$ 时刻的节点和边的集合。在动态异质信息网络中，每个节点 $v \in \mathcal{V}$ 和边 $e \in \mathcal{E}$  分别关联一个类型映射函数 $\phi: \mathcal{V} \rightarrow \mathcal{O}$ 和 $\varphi: \mathcal{E} \rightarrow \mathcal{R}$ ，其中 $\mathcal{O}$ 和 $\mathcal{R}$  表示节点和边的类型集合。当 $|\mathcal{O}| + |\mathcal{R}| > 2$  时，图 $\mathcal{G}$ 称为动态异质信息网络。

#### 定义 2：基于元路径的一阶相似性

对于一个节点对 $(v_i, v_j)$ ，在元路径 $m$ 下，连接该节点对的路径实例个数表示节点 $v_i$ 和 $v_j$ 间的一阶相似性，其度量异质信息网络中节点间的局部结构相似性。

#### 定义 3：基于元路径的二阶相似性

在元路径 $m$ 下，节点 $v_i$ 的邻居集合 $\mathcal{N}(v_i)^m$ 包含所有和 $v_i$ 通过元路径实例连接的节点。节点 $v_i$ 与其邻居 $\mathcal{N}(v_i)^m$ 之间的相似性定义为基于元路径 $m$ 的二阶相似性，其基于邻居结构度量了节点间的相似性。

图 5-1 展示了基于元路径的一阶相似性和二阶相似性。需要注意的是，基于元路径的一阶相似性表示异质信息网络中节点间的点对相似性，而基于元路径的二阶相似性定义了节点和其邻居集合之间的相似性。

#### 定义 4：元路径增强的邻接矩阵

给定一条元路径 $m$ ，元路径增强邻接矩阵定义为 $\mathbf{W}^m = [w_{ij}^m]$ ，其中 $w_{ij}^m$ 表示沿元路径 $m$ 连接节点 $v_i$ 和节点 $v_j$ 的路径实例的数量。自然地，矩阵 $\mathbf{W}^m$ 结合了异质信息网络的拓扑结构和语义信息。当 $m$ 是对称元路径时，矩阵 $\mathbf{W}^m$ 也是对称的。

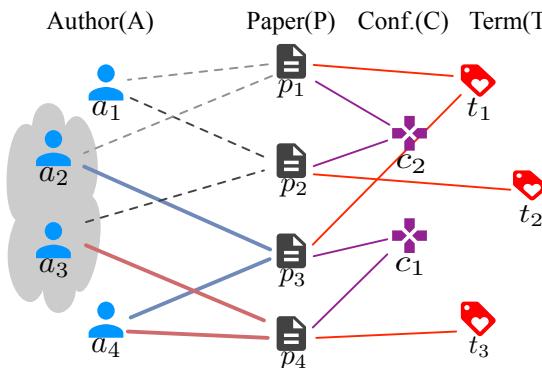


图 5-1 异质信息网络中基于元路径的一阶和二阶相似性的示例。基于元路径 APA 时，在低维空间中由元路径实例 $a_1p_2a_2$ 连接的节点 $a_1$ 和 $a_2$ 应该是接近的，这表示基于元路径的一阶相似性。由于 $a_1$ 的邻居与 $a_4$ 的邻居相同，即 $\{a_2, a_3\}$ ，因此在元路径 APA 下，节点 $a_1$ 和 $a_4$ 在低维位置也应彼此靠近即使它们没有直接连接，这表示基于元路径的二阶相似性。

### 5.2.2 基本思路

该研究所提模型 DyHNE 的核心思想是建立一个有效且高效的架构，该架构可以捕获动态异质信息网中拓扑结构和语义的变化并有效地学习节点嵌入表示。为了实现这一点，该研究首先引入基于元路径的一阶和二阶相似性，以保留异质信息网络中的结构和语义。如图 5-2 所示，基于元路径 APA, APCPA 和 APTPA 的三个增强邻接矩阵被定义并赋予权重进行融合，从而在  $t$  时刻获得融合矩阵  $\mathbf{W}^{(t)}$ 。然后，本论文提出了一个基础的静态异质信息网络表示学习模型 (StHNE)，该模型通过解决融合矩阵  $\mathbf{W}^{(t)}$  的广义特征值问题来学习在  $t$  时刻的节点嵌入表示  $\mathbf{U}^{(t)}$ 。随着异质信息网络从  $t$  时刻演化到  $t+1$  时刻，新的节点和边被添加到网络中（即节点  $a_3, p_4$  和  $t_3$ ；边  $(a_3, p_4), (a_1, p_4), (p_4, c_2), (p_4, t_2)$  和  $(p_4, t_3)$ ），这同时导致了元路径增强邻接矩阵的变化。由于这些矩阵实际上建模了异质信息网络的拓扑结构和语义，因此可以通过融合矩阵的扰动（即  $\Delta \mathbf{W}$ ）捕获动态异质信息网络的结构和语义变化。此外，利用矩阵扰动理论，可以构建专门针对动态异质信息网络嵌入表示的更新方程，以便所提模型 DyHNE 能够有效地计算得出变化的嵌入表示  $\Delta \mathbf{U}$ ，并根据公式  $\mathbf{U}^{(t+1)} = \mathbf{U}^{(t)} + \Delta \mathbf{U}$  将网络嵌入表示从  $\mathbf{U}^{(t)}$  更新到  $\mathbf{U}^{(t+1)}$ 。

简而言之，所提出的 StHNE 能够捕获异质信息网络中基于元路径的一阶和二阶相似性的拓扑结构和语义信息，并且 DyHNE 模型通过扰乱元路径增强的邻接矩阵来实现网络嵌入表示的有效更新。

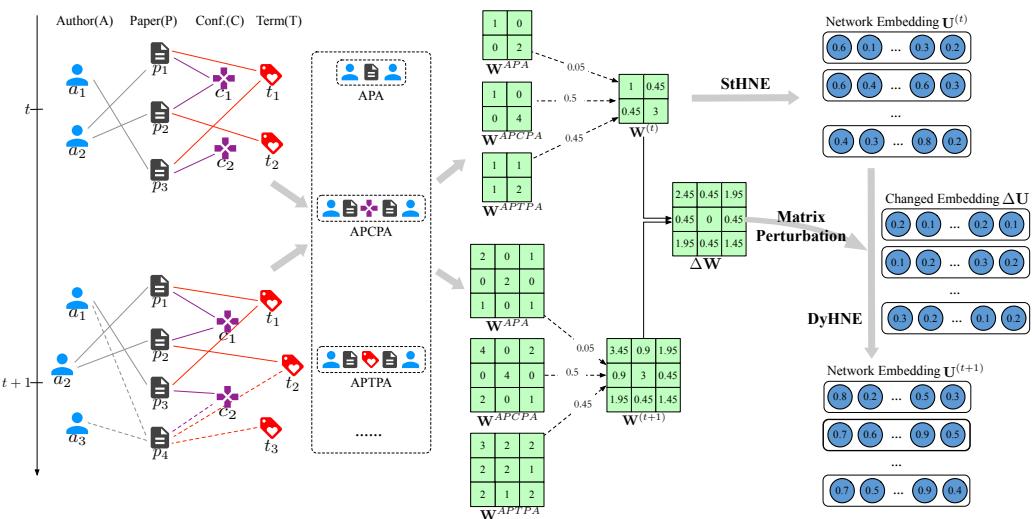


图 5-2 StHNE 和 DyHNE 模型结构

### 5.2.3 静态异质信息网络表示学习模型

在实现有效的动态更新节点嵌入之前，必须有适当的静态异质信息网络表示学习模型来捕获拓扑结构和语义信息。因此，本论文首先提出一个静态的异质信息网络表示学习模型 (StHNE)，该模型保留了基于元路径的一阶和二阶相似性。

### 5.2.3.1 保持基于元路径的一阶相似性

基于元路径的一阶相似性建模异质信息网络中节点间的局部结构，给定节点对 $(v_i, v_j)$ 由元路径 $m$ 的路径实例连接，基于元路径的一阶相似性建模如下：

$$p_1^m(v_i, v_j) = w_{ij}^m \|\mathbf{u}_i - \mathbf{u}_j\|_2^2 \quad (5-1)$$

其中 $\mathbf{u}_i \in \mathbb{R}^d$ 是节点 $v_i$ 的 $d$ -维向量表示。模型最小化如下的目标函数：

$$\mathcal{L}_1^m = \sum_{v_i, v_j \in V} p_1^m(v_i, v_j) = \sum_{v_i, v_j \in V} w_{ij}^m \|\mathbf{u}_i - \mathbf{u}_j\|_2^2 \quad (5-2)$$

由于较大的 $w_{ij}^m$ 表示节点 $v_i$ 和节点 $v_j$ 沿着元路径 $m$ 具有更多的连接，这应该使得节点 $v_i$ 和节点 $v_j$ 在低维表示空间中也更接近。

### 5.2.3.2 保持基于元路径的二阶相似性

基于元路径的二阶相似性取决于节点间共享的邻居结构。给定节点 $v_p$ 在元路径 $m$ 下的邻居节点集合 $\mathcal{N}(v_p)^m$ ，可以建模基于元路径的二阶相似性如下：

$$p_2^m(v_p, \mathcal{N}(v_p)^m) = \|\mathbf{u}_p - \sum_{v_q \in \mathcal{N}(v_p)^m} w_{pq}^m \mathbf{u}_q\|_2^2 \quad (5-3)$$

这里对矩阵 $w_{pq}^m$ 进行了归一化，以使得 $\sum_{v_q \in \mathcal{N}(v_p)^m} w_{pq}^m \mathbf{u}_q = 1$ 。

以上定义保证了节点 $v_p$ 和其沿元路径可达的邻居是相似的。如图 5-1 所示，在元路径 APA 下，节点 $a_1$ 和 $a_4$ 存在共同的邻居集合 $\{a_2, a_3\}$ 。公式(5-3)的定义，使得节点 $a_1$ 在嵌入空间内尽量靠近集合 $\{a_2, a_3\}$ 。同样的，在低维隐含空间内，节点 $a_4$ 和集合 $\{a_2, a_3\}$ 也是接近的。因此即使节点 $a_1$ 和节点 $a_4$ 没有直接连接，但两者也是相似的。如上述公式定义的，这隐式保留了两个未连接节点的基于元路径的二阶相似性。为了保留基于元路径的二阶相似性，最小化如下的目标函数：

$$\mathcal{L}_2^m = \sum_{v_p \in \mathcal{V}} \|\mathbf{u}_p - \sum_{v_q \in \mathcal{N}(v_p)^m} w_{pq}^m \mathbf{u}_q\|_2^2 \quad (5-4)$$

直观地，最小化上述目标函数，将在低维隐含空间中使节点 $v_p$ 与它的邻居之间的距离很小。因此，与节点 $v_p$ 存在相同邻居的节点也将接近 $v_p$ 。通过这种方式，网络的嵌入表示能够编码异质信息网络中基于元路径的二阶相似性。

### 5.2.3.3 统一的静态模型

考虑到异质信息网络中的多类型语义关系，接下来定义元路径集合 $\mathcal{M}$ 并且为集合中的每条元路径分别赋予权重 $\{\theta_1, \theta_2, \theta_3, \dots, \theta_{|\mathcal{M}|}\}$ ， $\sum_{i=1}^{|\mathcal{M}|} \theta_i = 1$ 。通过融合多条元路径，同时保留基于元路径的一阶和二阶相似性，可以定义如下的模型：

$$\mathcal{L} = \sum_{m \in \mathcal{M}} \theta_m (\mathcal{L}_1^m + \gamma \mathcal{L}_2^m) \quad (5-5)$$

其中， $\gamma$ 是一阶和二阶相似性之间的平衡系数。至此，静态异质信息网络表示学习问题可以转化为如下：

$$\arg \min_{\mathbf{U}^\top \mathbf{D} \mathbf{U} = \mathbf{I}} \sum_{m \in \mathcal{M}} \theta_m (\mathcal{L}_1^m + \gamma \mathcal{L}_2^m) \quad (5-6)$$

其中  $\mathbf{U}^\top \mathbf{D} \mathbf{U} = \mathbf{I}$  消除任意缩放因子，避免所有节点的表示向量是相同的。

### 5.2.4 基于谱理论的优化

受谱理论<sup>[41,85]</sup>的启发，可以将上述异质信息网络表示学习问题转变为求解广义特征值问题，这样可以得到一个闭式解，同时可以利用特征摄动理论<sup>[85]</sup>高效动态地更新节点的表示。根据谱理论，可以将公式(5-2)重新写作：

$$\mathcal{L}_1^m = \sum_{v_i, v_j \in \mathcal{V}} w_{ij}^m \| \mathbf{u}_i - \mathbf{u}_j \|_2^2 = 2 \operatorname{tr} (\mathbf{U}^\top \mathbf{L}^m \mathbf{U}) \quad (5-7)$$

其中  $\operatorname{tr}(\cdot)$  表示矩阵的迹， $\mathbf{L}^m = \mathbf{D}^m - \mathbf{W}^m$  是元路径  $m$  下的拉普拉斯矩阵， $\mathbf{D}^m$  是对角矩阵，其中  $\mathbf{D}_{ii}^m = \sum_j w_{ij}^m$ 。类似的，公式(5-4)可以被重新写为如下形式：

$$\mathcal{L}_2^m = \sum_{v_p \in \mathcal{V}} \| \mathbf{u}_p - \sum_{v_q \in \mathcal{N}(v_p)^m} w_{pq}^m \mathbf{u}_q \|_2^2 = 2 \operatorname{tr} (\mathbf{U}^\top \mathbf{H}^m \mathbf{U}) \quad (5-8)$$

其中， $\mathbf{H}^m = (\mathbf{I} - \mathbf{W}^m)^T (\mathbf{I} - \mathbf{W}^m)$  是一个对称矩阵。正如前面讨论过的，算法融合元路径集合  $\mathcal{M}$  中的所有元路径，即：

$$\mathbf{W} = \sum_{m \in \mathcal{M}} \theta_m \mathbf{W}^m, \quad \mathbf{D} = \sum_{m \in \mathcal{M}} \theta_m \mathbf{D}^m \quad (5-9)$$

因此，静态异质信息网络表示学习模型 StHNE 可以重新组织为：

$$\mathcal{L} = \operatorname{tr} (\mathbf{U}^\top (\mathbf{L} + \gamma \mathbf{H}) \mathbf{U}) \quad (5-10)$$

其中  $\mathbf{L} = \mathbf{W} - \mathbf{D}$ ， $\mathbf{H} = (\mathbf{I} - \mathbf{W})^T (\mathbf{I} - \mathbf{W})$ 。至此，StHNE 模型可以表示为：

$$\arg \min_{\mathbf{U}^\top \mathbf{D} \mathbf{U} = \mathbf{I}} \operatorname{tr} (\mathbf{U}^\top (\mathbf{L} + \gamma \mathbf{H}) \mathbf{U}) \quad (5-11)$$

其中  $\mathbf{L} + \gamma \mathbf{H}$  是对称的。公式(5-11)的问题可以转化为如下的广义特征值问题<sup>[86]</sup>：

$$(\mathbf{L} + \gamma \mathbf{H}) \mathbf{U} = \mathbf{D} \boldsymbol{\Lambda} \mathbf{U} \quad (5-12)$$

其中  $\boldsymbol{\Lambda} = \operatorname{diag}(\lambda_1, \lambda_2, \dots, \lambda_{N_M})$  是特征向量矩阵， $N_M$  是元路径集合  $\mathcal{M}$  中节点数量。

至此，模型 StHNE 转化为广义特征值问题，表示矩阵  $\mathbf{U}$  可以通过前  $d$  个非零特征值对应的特征向量给出。因此，通过求解广义特征值问题，可以得到网络在  $t$  时刻的表示矩阵  $\mathbf{U}^{(t)}$ 。当异质信息网络在  $t + 1$  时刻发生改变时，动态异质信息网络表示学习模型聚焦于高效更新  $\mathbf{U}^{(t)}$  到  $\mathbf{U}^{(t+1)}$ ，也就是更新特征值和特征向量。

### 5.2.5 动态异质信息网络表示学习模型

动态异质信息网络表示学习模型的核心思想是：以动态的方式，高效学习节点的表示。因此，下一步基于矩阵摄动，构建高效更新特征值和特征向量的方法。

#### 5.2.5.1 矩阵扰动构建动态网络嵌入表示

根据先前的工作<sup>[32,65]</sup>，假设网络的最终节点数量为  $N$ 。对于当前某时刻不存在的节点，将其视为度为 0 的孤立节点，因此网络的演变可以看作是边连接的变

化<sup>[59]</sup>。此外，边连接的增加（删除）会因边类型的不同而有所差异。因此，元路径增强的邻接矩阵的变化 $\Delta\mathbf{W}^m$ 可以自然地反映出网络结构和语义的变化。因为摄动理论可以通过添加扰动项来给出问题的近似解，所以可以利用特征值摄动更新特征值和特征向量。具体地，我们将元路径增强的连接矩阵摄动记为 $\Delta\mathbf{W} = \sum_{m \in \mathcal{M}} \theta_m \Delta\mathbf{W}^m$ 。因此，对于矩阵 $\mathbf{L}$ 和矩阵 $\mathbf{H}$ 的变化定义为如下：

$$\Delta\mathbf{L} = \Delta\mathbf{D} - \Delta\mathbf{W} \quad (5-13)$$

$$\Delta\mathbf{H} = \Delta\mathbf{W}^\top \Delta\mathbf{W} - (\mathbf{I} - \mathbf{W})^\top \Delta\mathbf{W} - \Delta\mathbf{W}^\top (\mathbf{I} - \mathbf{W}) \quad (5-14)$$

由于摄动理论可以通过添加一个微扰项来近似解决问题<sup>[85]</sup>，因此可以利用特征值微扰从上一次的特征值和特征向量中更新特征值和特征向量。因此，在新的时间步上，基于公式(5-12)可以得到：

$$\begin{aligned} & (\mathbf{L} + \Delta\mathbf{L} + \gamma\mathbf{H} + \gamma\Delta\mathbf{H})(\mathbf{U} + \Delta\mathbf{U}) \\ &= (\mathbf{D} + \Delta\mathbf{D})(\mathbf{\Lambda} + \Delta\mathbf{\Lambda})(\mathbf{U} + \Delta\mathbf{U}) \end{aligned} \quad (5-15)$$

其中 $\Delta\mathbf{U}$ 和 $\Delta\mathbf{\Lambda}$ 表示特征值和特征向量的变化量。因为每个时间步上的扰动过程是相同的，所以这里为了简洁省略了上标( $t$ )。现在，聚焦于一个特定的特征对 $(\mathbf{u}_i, \lambda_i)$ ，公式(5-15)可以重新写作如下形式：

$$\begin{aligned} & (\mathbf{L} + \Delta\mathbf{L} + \gamma\mathbf{H} + \gamma\Delta\mathbf{H})(\mathbf{u}_i + \Delta\mathbf{u}_i) \\ &= (\lambda_i + \Delta\lambda_i)(\mathbf{D} + \Delta\mathbf{D})(\mathbf{u}_i + \Delta\mathbf{u}_i) \end{aligned} \quad (5-16)$$

因此，动态异质信息网络表示学习问题转换成：如何计算第 $i$ 个特征对 $(\Delta\mathbf{u}_i, \Delta\lambda_i)$ 的变化。因为如果计算得到在 $t$ 和 $t+1$ 时刻之间的变化量 $\Delta\mathbf{U}$ 和 $\Delta\mathbf{\Lambda}$ ，则可以有效地利用公式 $\mathbf{U}^{(t+1)} = \mathbf{U}^{(t)} + \Delta\mathbf{U}$ 更新网络嵌入表示矩阵。

接下来，首先介绍如何计算 $\Delta\lambda_i$ 。展开公式(5-16)并移除对结果影响不大的高阶项<sup>[87]</sup>，例如 $\Delta\mathbf{L}\Delta\mathbf{u}_i$ 和 $\Delta\lambda_i\Delta\mathbf{D}\Delta\mathbf{u}_i$ ，然后基于等式 $(\mathbf{L} + \gamma\mathbf{H})\mathbf{u}_i = \lambda_i\mathbf{D}\mathbf{u}_i$ ，有：

$$\begin{aligned} & (\mathbf{L} + \gamma\mathbf{H})\Delta\mathbf{u}_i + (\Delta\mathbf{L} + \gamma\Delta\mathbf{H})\mathbf{u}_i \\ &= \lambda_i\mathbf{D}\Delta\mathbf{u}_i + \lambda_i\Delta\mathbf{D}\mathbf{u}_i + \Delta\lambda_i\mathbf{D}\mathbf{u}_i \end{aligned} \quad (5-17)$$

进一步地，对上式左乘 $\mathbf{u}_i^\top$ ，可以得到：

$$\begin{aligned} & \mathbf{u}_i^\top (\mathbf{L} + \gamma\mathbf{H})\Delta\mathbf{u}_i + \mathbf{u}_i^\top (\Delta\mathbf{L} + \gamma\Delta\mathbf{H})\mathbf{u}_i \\ &= \lambda_i \mathbf{u}_i^\top \mathbf{D}\Delta\mathbf{u}_i + \lambda_i \mathbf{u}_i^\top \Delta\mathbf{D}\mathbf{u}_i + \Delta\lambda_i \mathbf{u}_i^\top \mathbf{D}\mathbf{u}_i \end{aligned} \quad (5-18)$$

由于 $\mathbf{L} + \gamma\mathbf{H}$ 和 $\mathbf{D}$ 是对称的，对等式 $(\mathbf{L} + \gamma\mathbf{H})\mathbf{u}_i = \lambda_i\mathbf{D}\mathbf{u}_i$ 右乘 $\Delta\mathbf{u}_i$ 可以得到 $\mathbf{u}_i^\top (\mathbf{L} + \gamma\mathbf{H})\Delta\mathbf{u}_i = \lambda_i \mathbf{u}_i^\top \mathbf{D}\Delta\mathbf{u}_i$ 。因此公式(5-18)可以重写为：

$$\mathbf{u}_i^\top (\Delta\mathbf{L} + \gamma\Delta\mathbf{H})\mathbf{u}_i = \lambda_i \mathbf{u}_i^\top \Delta\mathbf{D}\mathbf{u}_i + \Delta\lambda_i \mathbf{u}_i^\top \mathbf{D}\mathbf{u}_i \quad (5-19)$$

基于公式(5-19)，可以得到特征值的变化量：

$$\Delta\lambda_i = \frac{\mathbf{u}_i^\top \Delta\mathbf{L}\mathbf{u}_i + \gamma \mathbf{u}_i^\top \Delta\mathbf{H}\mathbf{u}_i - \lambda_i \mathbf{u}_i^\top \Delta\mathbf{D}\mathbf{u}_i}{\mathbf{u}_i^\top \mathbf{D}\mathbf{u}_i} \quad (5-20)$$

因为 $\mathbf{D}$ 是半正定矩阵，可以得到 $\mathbf{u}_i^\top \mathbf{D}\mathbf{u}_i = 1$ 和 $\mathbf{u}_i^\top \mathbf{D}\mathbf{u}_j = 0 (i \neq j)$ <sup>[88]</sup>，因此：

$$\Delta\lambda_i = \mathbf{u}_i^\top \Delta\mathbf{L}\mathbf{u}_i + \gamma \mathbf{u}_i^\top \Delta\mathbf{H}\mathbf{u}_i - \lambda_i \mathbf{u}_i^\top \Delta\mathbf{D}\mathbf{u}_i \quad (5-21)$$

得到了两个连续时间步间的特征值变化量 $\Delta\lambda_i$ , 接下来可以计算特征向量的变化量 $\Delta\mathbf{u}_i$ 。因为异质信息网络的演变是平缓的<sup>[10,60]</sup>, 基于元路径的网络变化(即 $\Delta\mathbf{W}$ )是微小的。根据特征向量变化分析<sup>[85]</sup>, 有:

$$\Delta\mathbf{u}_i = \sum_{j=2,j \neq i}^{d+1} \alpha_{ij} \mathbf{u}_j \quad (5-22)$$

其中,  $\alpha_{ij}$ 表示第 $j$ 个特征向量 $\mathbf{u}_j$ 对 $\Delta\mathbf{u}_i$ 的权重。因此, 计算 $\Delta\mathbf{u}_i$ 的问题现在转变为如何决定这些权重值。考虑公式(5-16), 通过将所有的 $\Delta\mathbf{u}_i$ 替换为公式(5-18), 同时约减对结果影响很小的高阶项<sup>[87]</sup>, 可以得到如下公式:

$$\begin{aligned} & (\mathbf{L} + \gamma\mathbf{H}) \sum_{j=2,j \neq i}^{d+1} \alpha_{ij} \mathbf{u}_j + (\Delta\mathbf{L} + \gamma\Delta\mathbf{H})\mathbf{u}_i \\ &= \lambda_i \mathbf{D} \sum_{j=2,j \neq i}^{d+1} \alpha_{ij} \mathbf{u}_j + \lambda_i \Delta\mathbf{D}\mathbf{u}_i + \Delta\lambda_i \mathbf{D}\mathbf{u}_i \end{aligned} \quad (5-23)$$

基于等式 $(\mathbf{L} + \gamma\mathbf{H}) \sum_{j=2}^{d+1} \alpha_{ij} \mathbf{u}_j = \mathbf{D} \sum_{j=2}^{d+1} \alpha_{ij} \lambda_j \mathbf{u}_j$ , 并上式两边同乘 $\mathbf{u}_p^\top$  $(2 \leq p \leq d+1, p \neq i)$ , 可以得到:

$$\begin{aligned} & \mathbf{u}_p^\top \mathbf{D} \sum_{j=2,j \neq i}^{d+1} \alpha_{ij} \lambda_j \mathbf{u}_j + \mathbf{u}_p^\top (\Delta\mathbf{L} + \gamma\Delta\mathbf{H})\mathbf{u}_i \\ &= \lambda_i \mathbf{u}_p^\top \mathbf{D} \sum_{j=2,j \neq i}^{d+1} \alpha_{ij} \mathbf{u}_j + \lambda_i \mathbf{u}_p^\top \Delta\mathbf{D}\mathbf{u}_i + \Delta\lambda_i \mathbf{u}_p^\top \mathbf{D}\mathbf{u}_i \end{aligned} \quad (5-24)$$

基于等式 $\mathbf{u}_i^\top \mathbf{D}\mathbf{u}_i = 1$ 和 $\mathbf{u}_i^\top \mathbf{D}\mathbf{u}_j = 0(i \neq j)$ , 可以简化上式得到:

$$\lambda_p \alpha_{ip} + \mathbf{u}_p^\top (\Delta\mathbf{L} + \gamma\Delta\mathbf{H})\mathbf{u}_i = \lambda_i \alpha_{ip} + \lambda_i \mathbf{u}_p^\top \Delta\mathbf{D}\mathbf{u}_i \quad (5-25)$$

最后, 通过上式可以得到权重 $\alpha_{ip}$ :

$$\alpha_{ip} = \frac{\mathbf{u}_p^\top \Delta\mathbf{L}\mathbf{u}_i + \gamma \mathbf{u}_p^\top \Delta\mathbf{H}\mathbf{u}_i - \lambda_i \mathbf{u}_p^\top \Delta\mathbf{D}\mathbf{u}_i}{\lambda_i - \lambda_p}, i \neq p \quad (5-26)$$

总结一下, 至此基于公式(5-21)(5-22)和(5-26)计算得到了特征值和特征向量的变化量。因此, 在 $t+1$ 时刻, 新的特征值和特征向量可以以如下方式更新:

$$\boldsymbol{\Lambda}^{(t+1)} = \boldsymbol{\Lambda}^{(t)} + \Delta\boldsymbol{\Lambda}, \quad \mathbf{U}^{(t+1)} = \mathbf{U}^{(t)} + \Delta\mathbf{U} \quad (5-27)$$

### 5.2.5.2 算法加速

至此, 更新异质信息网络嵌入表示的一个简单方法是直接根据公式(5-21)(5-22)和(5-26)计算公式(5-27)。但 $\Delta\mathbf{H}$ 的定义导致公式(5-21)的计算复杂度非常高, 因此, 接下来本研究特别设计了针对动态异质信息网络表示学习的加速算法。

下面更详细地聚焦于 $\Delta\lambda_i$ 和 $\alpha_{ij}$ 的计算。利用公式(5-14)替换 $\Delta\mathbf{H}$ 并且约减高阶项, 公式(5-21)和(5-26)可以重新写为:

$$\begin{aligned}\Delta\lambda_i &= \mathbf{u}_i^\top \Delta\mathbf{L}\mathbf{u}_i - \lambda_i \mathbf{u}_i^\top \Delta\mathbf{D}\mathbf{u}_i \\ &\quad + \gamma \left\{ [(\mathbf{W} - \mathbf{I})\mathbf{u}_i]^\top \Delta\mathbf{W}\mathbf{u}_i + (\Delta\mathbf{W}\mathbf{u}_i)^\top (\mathbf{W} - \mathbf{I})\mathbf{u}_i \right\}\end{aligned}\quad (5-28)$$

$$\begin{aligned}\alpha_{ij} &= \frac{\mathbf{u}_j^\top \Delta\mathbf{L}\mathbf{u}_i - \lambda_i \mathbf{u}_j^\top \Delta\mathbf{D}\mathbf{u}_i}{\lambda_i - \lambda_j} \\ &\quad + \frac{\gamma \left\{ [(\mathbf{W} - \mathbf{I})\mathbf{u}_j]^\top \Delta\mathbf{W}\mathbf{u}_i + (\Delta\mathbf{W}\mathbf{u}_j)^\top (\mathbf{W} - \mathbf{I})\mathbf{u}_i \right\}}{\lambda_i - \lambda_j}\end{aligned}\quad (5-29)$$

为了简便表示，上式可以写作：

$$\Delta\lambda_i = \mathbf{C}(i, i) + \gamma [\mathbf{A}(:, i)^\top \mathbf{B}(:, i) + \mathbf{B}(:, i)^\top \mathbf{A}(:, i)] \quad (5-30)$$

$$\alpha_{ij} = \frac{\mathbf{C}(j, i) + \gamma [\mathbf{A}(i, j)^\top \mathbf{B}(:, i) + \mathbf{B}(:, j)^\top \mathbf{A}(i, i)]}{\lambda_i - \lambda_j} \quad (5-31)$$

其中  $\mathbf{A}(:, i) = (\mathbf{W} - \mathbf{I})\mathbf{u}_i$ ,  $\mathbf{B}(:, i) = \Delta\mathbf{W}\mathbf{u}_i$ , 并且  $\mathbf{C}(i, j) = \mathbf{u}_i^\top \Delta\mathbf{L}\mathbf{u}_j - \lambda_i \mathbf{u}_i^\top \Delta\mathbf{D}\mathbf{u}_j$ 。很显然，对于  $\mathbf{A}$  的计算时非常耗时的。因此，可以将  $t+1$  时刻的  $\mathbf{A}^{(t+1)}(:, i)$  定义为：

$$\mathbf{A}^{(t+1)}(:, i) = (\mathbf{W} - \mathbf{I} + \Delta\mathbf{W})(\mathbf{u}_i + \Delta\mathbf{u}_i) \quad (5-32)$$

将  $\Delta\mathbf{u}_i$  替换为公式(5-22)，可以得到：

$$\begin{aligned}\mathbf{A}^{(t+1)}(:, i) &= (\mathbf{W} - \mathbf{I} + \Delta\mathbf{W}) \left( \mathbf{u}_i + \sum_{j=2, j \neq i}^{d+1} \alpha_{ij} \mathbf{u}_j \right) \\ &= \sum_{j=2}^{d+1} \beta_{ij} (\mathbf{W} - \mathbf{I} + \Delta\mathbf{W}) \mathbf{u}_j\end{aligned}\quad (5-33)$$

其中，当  $i \neq j$  时， $\beta_{ij} = \alpha_{ij}$ ；相反地， $\beta_{ij} = 1$ 。更进一步地，可以得到如下公式：

$$\mathbf{A}^{(t+1)}(:, i) = \sum_{j=2}^{d+1} \beta_{ij} (\mathbf{A}^t(:, j) + \mathbf{B}^t(:, j)) \quad (5-34)$$

基于公式(5-34)，可以将更新  $\mathbf{A}^{(t+1)}$  的时间复杂度从  $O(ed)$  降低到  $O(d^2)$ 。

### 5.2.5.3 算法复杂度分析

由于只需要在最开始的时候运行一次静态异质网络表示学习模型 StHNE，所以所提 DyHNE 算法可以高效地动态更新节点的表示。对于静态异质网络表示学习模型 StHNE，给定一个元路径集合  $\mathcal{M}$ ，其中包含  $N_{\mathcal{M}}$  个节点，求解广义特征值问题的时间复杂度为  $O(dN_{\mathcal{M}}^2)$ ，其中  $d$  是节点表示向量的维度。对于动态异质信息网络表示学习模型 DyHNE，假设总共有  $T$  个时间步， $f$  和  $g$  分别是矩阵  $\Delta\mathbf{D}$  和  $\Delta\mathbf{W}$  中的非零元素个数。在每个时间步，计算  $\mathbf{B}$  和  $\mathbf{C}$  的时间复杂度分别是  $O(fd)$  和  $O((f+g)d^2)$ 。为了计算  $\alpha_i$  和 top- $d$  特征向量，所提模型需要的时间复杂度分别是  $O(d^2)$  和  $O(d^2 N_{\mathcal{M}})$ 。整体上，在  $T$  个时间步上，本论文提出的动态更新算法 DyHNE 的时间复杂度为  $O(T(f+g+N_{\mathcal{M}})d^2)$ 。由于  $d \ll N_{\mathcal{M}}$ ,  $f$  和  $g$  通常很小，所以动态模型 DyHNE 的时间复杂度和异质信息网络中节点数量呈线性关系的。

## 5.3 实验及分析

### 5.3.1 实验设置

#### 5.3.1.1 数据集

该实验在三个数据集上评估模型，表 5-1 汇总了数据集的统计信息。

- Yelp 是 Yelp Challenge 提供的电商评论数据集。在本实验中提取与三个子类别的餐厅有关的信息：“美国食品”，“快餐”和“寿司吧”，并构建一个异质信息网络。在本实验中采用的元路径是 BRURB（即，用户对两个商家上进行过评论）和 BSB（即，同一星级的商家）。
- DBLP 是计算机领域的学术合作网络。在 DBLP 中，有 4057 位作者被标记了他们的研究领域，例如数据挖掘。本实验中考虑元路径，包括 APA（即合作关系），APCPA（即作者参加同一会议）和 APTPA（即作者使用同一术语）。
- AMiner 是学术合作网络，它包括从 1990 年到 2005 年的五个研究领域的论文信息。对于在这五个领域中发表的作者，他/她的标签被定义为其大部分出版物的类别。与 DBLP 一样，在本实验中采用元路径 APA，APCPA 和 APTPA。

表 5-1 数据集统计信息

Datasets	Node Types	#Nodes	Meta-path	Time Steps
Yelp	Star (S)	9	BSB BRURB	10
	User (U)	1,286		
	Review (R)	33,360		
	Business (B)	2,614		
DBLP	Term (T)	8,833	APA APCPA APTPA	10
	Paper (P)	14,376		
	Author (A)	14,475		
	Conference (C)	20		
AMiner	Term (T)	8,811	APA APCPA APTPA	10
	Paper (P)	18,181		
	Author (A)	22,942		
	Conference (C)	22		

#### 5.3.1.2 对比方法

在本实验中，所提模型 StHNE 和 DyHNE 对比了多种的网络表示学习方法，包括两个同质网络表示学习方法（即 DeepWalk<sup>[24]</sup>和 LINE<sup>[46]</sup>）；两个静态异质信息网络表示学习方法（即 ESim<sup>[53]</sup>和 metapath2vec<sup>[37]</sup>）；和两个动态的同质信息网络表示学习方法（即 DANE<sup>[65]</sup>和 DHPE<sup>[32]</sup>）。另外，为了验证基于元路径的一阶和二阶相似性的有效性，在本实验中测试了 StHNE-1st 和 StHNE-2nd 的性能。

#### 5.3.1.3 参数设置

为了公平地比较，本实验中所有模型的节点的向量表示维度设置为 100。在本研究所提模型 StHNE 和 DyHNE 中，一阶相似性和二阶相似性之间的平衡系

数  $\gamma$  设置为 1。对于元路径的权重，采用范围为(0,1)的网格搜索获得元路径的最佳加权组合。模型中负样本的大小设置为 5。对于基于随机游走的模型，每个节点的游走次数设置为 10，游走序列长度设置为 50，窗口大小设置为 5。对于无法处理异质信息网络的模型，本实验忽略节点类型和边类型。

### 5.3.2 StHNE 模型有效性实验

为了评估 StHNE 的有效性，该实验在整个演变后的网络上学习节点表示，而不考虑网络的动态演变。换言之，给定具有 10 个时间步长的动态网络  $\{G^1, \dots, G^{10}\}$ ，该实验在整个网络上，即  $G^1 \cup G^2 \cup \dots \cup G^{10}$  执行网络表示学习。

#### 5.3.2.1 节点分类

节点分类是评估网络表示学习性能的一项常见任务。在此任务中，在学习了完全进化网络上的节点嵌入表示之后，训练一个以节点嵌入表示作为输入特征的逻辑回归分类器。在该实验中，训练集的比率设置为 40%，60% 和 80%。对于 Yelp 数据集，元路径 BSB 和 BRURB 的权重设置为 0.4 和 0.6。在 DBLP 中，将权重 {0.05, 0.5, 0.45} 分别分配给元路径 {APA, APCPA, APTPA}。在 AMiner 中，为元路径 {APA, APCPA, APTPA} 分别分配权重 {0.25, 0.5, 0.25}。表 5-2 报告了关于 Macro-F1 和 Micro-F1 的实验结果。

表 5-2 在静态异质信息网络上节点分类实验结果

Datasets	Metric	Tr.Ratio	DeepWalk	LINE-1st	LINE-1st	ESim	metapath2vec	StHNE-1st	StHNE-2nd	StHNE
Yelp	Macro-F1	40%	0.6021	0.5389	0.5438	0.6387	0.5872	0.6193	0.5377	<b>0.6421</b>
		60%	0.5954	0.5865	0.5558	0.6464	0.6081	0.6639	0.5691	<b>0.6644</b>
		80%	0.6101	0.6012	0.6068	0.6793	0.6374	0.6909	0.5783	<b>0.6922</b>
	Micro-F1	40%	0.6520	0.6054	0.6105	0.6896	0.6427	0.6838	0.6118	<b>0.6902</b>
		60%	0.6472	0.6510	0.6233	0.7011	0.6681	0.7103	0.6309	<b>0.7017</b>
		80%	0.6673	0.6615	0.6367	0.7186	0.6875	0.7232	0.6367	<b>0.7326</b>
DBLP	Macro-F1	40%	0.9295	0.9271	0.9172	0.9354	0.9213	0.9392	0.9283	<b>0.9473</b>
		60%	0.9355	0.9298	0.9252	0.9362	0.9311	0.9436	0.9374	<b>0.9503</b>
		80%	0.9368	0.9273	0.9301	0.9451	0.9432	0.9511	0.9443	<b>0.9611</b>
	Micro-F1	40%	0.9331	0.9310	0.9219	0.9394	0.9228	0.9421	0.9312	<b>0.9503</b>
		60%	0.9383	0.9328	0.9291	0.9406	0.9305	0.9487	0.9389	<b>0.9519</b>
		80%	0.9392	0.9323	0.9347	0.9502	0.9484	0.9543	0.9496	<b>0.9643</b>
AMiner	Macro-F1	40%	0.8838	0.8929	0.8972	0.9449	<b>0.9487</b>	0.9389	0.9309	0.9452
		60%	0.8846	0.8909	0.8967	0.9482	0.9490	0.9401	0.9354	<b>0.9499</b>
		80%	0.8853	0.8947	0.8962	0.9491	0.9493	0.9412	0.9381	<b>0.9521</b>
	Micro-F1	40%	0.8879	0.8925	0.8958	0.9465	<b>0.9469</b>	0.9407	0.9412	0.9467
		60%	0.8881	0.8936	0.8960	0.9482	0.9497	0.9423	0.9431	<b>0.9509</b>
		80%	0.8882	0.8960	0.8962	0.9500	0.9511	0.9448	0.9423	<b>0.9529</b>

如我们所见，本研究所提模型 StHNE 在三个数据集上的表现均优于所有对比方法。在 Macro-F1 指标上，模型 StHNE 的分类性能平均提高了约 8.7%，这得益于元路径的加权集成和网络结构的保留。虽然模型 StHNE，ESim 和 metapath2vec 都对多条元路径进行了带权融合，但是 ESim 和 metapath2vec 在三个数据集上的性能稍差。这可能是由于元路径融合过程与模型优化分离所致，这

导致丢失了用于异质信息网络嵌入的多个关系之间的某些信息。我们还注意到，在大多数情况下，StHNE-1st 和 StHNE-2nd 均优于 LINE-1st 和 LINE-2nd，这表明在异质信息网络中基于元路径的一阶和二阶相似性的有效性。纵向来看，StHNE 在不同大小的训练数据上持续表现最佳，表明了模型的稳定性和鲁棒性。

### 5.3.2.2 关系预测

对于 DBLP 和 AMiner，本实验对合作关系（APA）进行预测。因此，通过随机隐藏 DBLP 中 20% 的 AP 来生成训练网络，而 AMiner 数据更丰富，所以随机隐藏 AMiner 中 40% 的 AP 来生成训练网络。对于 Yelp，预测一个人评论过的两个商店（BRURB），可用于为用户推荐业务。因此，该实验随机隐藏 20% 的 BR 以生成训练网络。对于 Yelp 数据集，元路径 BSB 和 BRURB 的权重设置为 0.4 和 0.6。在 DBLP 中，将权重 {0.9, 0.05, 0.05} 分别分配给 {APA, APCPA, APTPA}。在 AMiner 中，将权重 {0.4, 0.3, 0.3} 分别分配给 {APA, APCPA, APTPA}。在该实验中，使用 AUC、F1 和 Accuracy 评估测试网络上的预测性能。

表 5-3 报告了不同方法的比较结果。总体而言，可以看到模型 StHNE 在两个指标上比其他对比方法具有更好的关系预测性能。显著的性能提升表明所提出的 StHNE 模型有效地保留了异质信息网络中的结构信息。受益于基于元路径保留的二阶相似性，StHNE-2nd 的性能明显优于 StHNE-1st。原因是，较高阶的结构相似性更有助于保留异质信息网络中的复杂关系。

表 5-3 在静态异质信息网络上关系预测实验结果

Datasets	Metric	DeepWalk	LINE-1st	LINE-1st	ESim	metapath2vec	StHNE-1st	StHNE-2nd	StHNE
Yelp	AUC	0.7404	0.6553	0.7896	0.6651	0.8187	0.8046	0.8233	<b>0.8356</b>
	F1	0.6864	0.6269	0.7370	0.6361	0.7355	0.7348	0.7397	<b>0.7512</b>
	ACC	0.6819	0.6115	0.7326	0.6386	0.7436	0.7286	0.7526	<b>0.7661</b>
DBLP	AUC	0.9235	0.8368	0.7672	0.9074	0.9291	0.9002	0.9246	<b>0.9385</b>
	F1	0.8424	0.7680	0.7054	0.8321	0.8645	0.8359	0.8631	<b>0.8850</b>
	ACC	0.8531	0.7680	0.6805	0.8416	0.8596	0.8266	0.8577	<b>0.8751</b>
AMiner	AUC	0.7366	0.5163	0.5835	0.8691	0.8783	0.8935	<b>0.9180</b>	0.8939
	F1	0.5209	0.5012	0.5276	0.6636	0.6697	0.7037	<b>0.8021</b>	0.7085
	ACC	0.6686	0.6475	0.6344	0.7425	0.7506	0.7622	<b>0.8251</b>	0.7701

### 5.3.3 DyHNE 模型有效性实验

该实验的目的是比较 DyHNE 与对比模型，以验证 DyHNE 的有效性。由于部分对比方法（例如 DeepWalk, LINE 和 StHNE）无法处理动态网络，并且已在 5.3.2 节中报告了这些方法的性能，因此这里仅将这些方法用于学习初始网络的节点表示。具体来说，给定一个具有 10 个时间步长的动态网络  $\{G^1, \dots, G^{10}\}$ ，对于包括 StHNE 在内的静态网络方法，仅对  $G^1$  进行学习并报告结果，而对于动态

网络表示学习方法，即 DANE, DHPE 和 DyHNE，本实验从网络  $\mathcal{G}^1$  到网络  $\mathcal{G}^{10}$  进行学习节点的动态表示，并报告最终结果以评估它们在动态场景中的性能。

### 5.3.3.1 节点分类

对于每个数据集，首先从原始网络中生成初始网络和增长网络，每个增长异质信息网络包含十个时间步长。在 Yelp 中，由于评论带有时间戳信息，在每个时间步向初始网络中随机添加 0.1% 的新 UR 和 BR 边连接。对于 DBLP，在每个时间步将 0.1% 的新 PA, PC 和 PT 边连接随机添加到初始网络。由于 AMiner 本身包含每篇论文的出版年份，因此将 2005 年的网络平均分为 10 个时间步长。

类似于 5.3.2.1 小节的实验设置，本实验将训练集的大小从 40% 更改为 80%，步长为 20%，其余节点作为测试。对每个分类实验重复十次，并报告 Macro-F1 和 Micro-F1 分数的平均表现，如表 5-4 所示。可以发现，在所有大小各异的数据集上，本论文所提出的 DyHNE 的性能始终优于其他对比方法，这证明了当用作节点分类的特征时，DyHNE 模型学习的节点嵌入表示的有效性和鲁棒性。特别地，DyHNE 显著优于两种动态网络表示学习方法 DANE 和 DHPE。原因是 DyHNE 模型考虑了节点和关系的不同类型，并且可以捕获异质信息网络中的拓扑结构和语义信息。与静态异质信息网络表示学习方法（例如 DeepWalk, LINE, ESim 和 metapath2vec）相比，DyHNE 也达到了最佳性能，这证明了更新算法的有效性，而又不会丢失异质信息网络中的重要结构和语义信息。

表 5-4 在动态异质信息网络上的节点分类实验结果

Datasets	Metric	Tr.Ratio	DeepWalk	LINE-1st	LINE-1st	ESim	metapath2vec	StHNE	DANE	DHPE	DyHNE
Yelp	Macro-F1	40%	0.5840	0.5623	0.5248	0.6463	0.5765	0.6118	0.6102	0.5412	<b>0.6459</b>
		60%	0.5962	0.5863	0.5392	0.6642	0.6192	0.6644	0.6342	0.5546	<b>0.6641</b>
		80%	0.6044	0.6001	0.6030	0.6744	0.6285	0.6882	0.6471	0.5616	<b>0.6893</b>
	Micro-F1	40%	0.6443	0.6214	0.5901	0.6932	0.6457	0.6826	0.6894	0.5823	<b>0.6933</b>
		60%	0.6558	0.6338	0.5435	0.6941	0.6656	0.7074	0.6921	0.5981	<b>0.6998</b>
		80%	0.6634	0.6424	0.6297	0.7104	0.6722	0.7281	0.6959	0.6034	<b>0.7298</b>
DBLP	Macro-F1	40%	0.9269	0.9266	0.9147	0.9372	0.9162	0.9395	0.8862	0.8893	<b>0.9434</b>
		60%	0.9297	0.9283	0.9141	0.9369	0.9253	0.9461	0.8956	0.8946	<b>0.9476</b>
		80%	0.9322	0.9291	0.9217	0.9376	0.9302	0.9502	0.9051	0.9087	<b>0.9581</b>
	Micro-F1	40%	0.9375	0.9310	0.9198	0.9383	0.9254	0.9438	0.8883	0.8847	<b>0.9467</b>
		60%	0.9346	0.9245	0.9192	0.9404	0.9281	0.9496	0.8879	0.8931	<b>0.9505</b>
		80%	0.9371	0.9297	0.9261	0.9415	0.9354	0.9543	0.9071	0.9041	<b>0.9617</b>
AMiner	Macro-F1	40%	0.8197	0.8219	0.8282	0.8797	0.8673	0.8628	0.7642	0.7694	<b>0.9014</b>
		60%	0.8221	0.8218	0.8323	0.8807	0.8734	0.8651	0.7704	0.7735	<b>0.9131</b>
		80%	0.8235	0.8238	0.8351	0.8821	0.8754	0.8778	0.7793	0.7851	<b>0.9212</b>
	Micro-F1	40%	0.8157	0.8189	0.8323	0.8729	0.8652	0.8563	0.7698	0.7633	<b>0.9117</b>
		60%	0.8175	0.8182	0.8361	0.8734	0.8693	0.8574	0.7723	0.7698	<b>0.9178</b>
		80%	0.8191	0.8201	0.8298	0.8751	0.8725	0.8728	0.7857	0.7704	<b>0.9203</b>

### 5.3.3.2 关系预测

对于每个数据集，首先从原始网络生成初始、增长和测试网络。对于 Yelp，首先建立包含 20% BR 边的测试网络。其余部分构成初始网络和增长网络，其中，增长网络分为 10 个时间步长，并且每个时间步长将 0.1% 的新 UR 和 BR 边连接

添加到初始网络中。对于 DBLP，使用同样方法。对于 AMiner，将 1990-2003 年的数据作为初始网络，2004 年的作为增长网络，2005 年的数据作为测试网络。

表 5-5 在动态异质信息网络上的关系预测实验结果

Datasets	Metric	DeepWalk	LINE-1st	LINE-1st	ESim	metapath2vec	StHNE	DANE	DHPE	DyHNE
Yelp	AUC	0.7316	0.6549	0.7895	0.6521	0.8164	0.8341	0.7928	0.7629	<b>0.8346</b>
	F1	0.6771	0.6125	0.7350	0.6168	0.7293	0.7506	0.7221	0.6809	<b>0.7504</b>
	ACC	0.6751	0.6059	0.7300	0.6185	0.7395	0.7616	0.7211	0.7023	<b>0.7639</b>
DBLP	AUC	0.9125	0.8261	0.7432	0.9053	0.9196	0.9216	0.5413	0.6411	<b>0.9278</b>
	F1	0.8421	0.7840	0.7014	0.8215	0.8497	0.8621	0.7141	0.6223	<b>0.8744</b>
	ACC	0.8221	0.7227	0.6754	0.8306	0.8405	0.8436	0.5511	0.5734	<b>0.8635</b>
AMiner	AUC	0.8660	0.6271	0.5648	0.8459	0.8694	0.8659	0.8405	0.8412	<b>0.8823</b>
	F1	0.7658	0.5651	0.6071	0.7172	0.7761	0.7567	0.7167	0.7158	<b>0.7792</b>
	ACC	0.7856	0.5328	0.5828	0.7594	0.7793	0.7733	0.7527	0.7545	<b>0.7889</b>

表 5-5 中报告了在 AUC、F1 和 Accuracy 指标上的性能。(1) 在三个数据集上，本文提出的 DyHNE 方法持续优于所有对比方法，这归因于基于元路径的一阶和二阶相似性保留了异质信息网络中的高阶拓扑结构；(2) 由于忽略了异质信息网络中的多种节点类型和关系类型，DANE 和 DHPE 的预测性能较差。

### 5.3.4 StHNE 模型效率性实验

在本节中，实验评估 StHNE 的效率。针对在三个数据集上的运行时间，将 StHNE 与其他静态网络表示学习模型进行比较，并以对数尺度进行绘制。

从图 5-3 中可以看出，尽管理论上 StHNE 的时间复杂度很高，但是 StHNE 仍然比其他方法快得多。与为静态网络设计的模型（例如 DeepWalk, LINE, ESim 和 metapath2vec）相比，StHNE 的运行时间与它们处于同一数量级，这是因为 StHNE 不需要迭代优化，只需要计算稀疏矩阵的 top-d 特征值和特征向量。

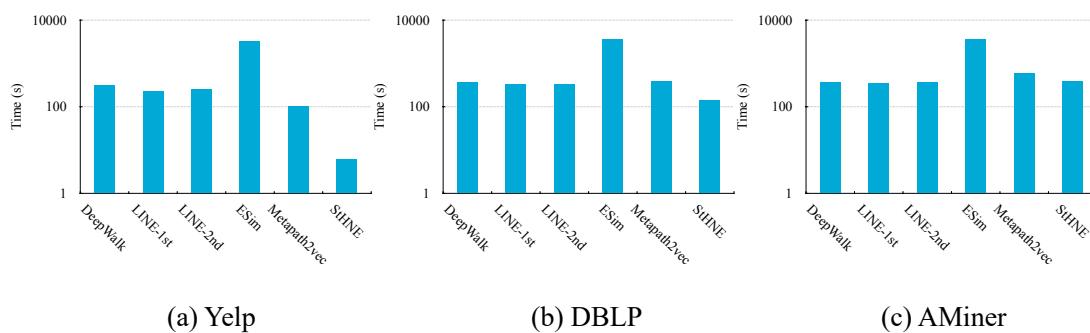


图 5-3 StHNE 有效性实验结果

### 5.3.5 DyHNE 模型效率性实验。

#### 5.3.5.1 和对比方法相比

由于 DANE 和 DHPE 也可以处理网络中的动态变化，因此在图 5-4 中比较了 DyHNE 与 DANE 和 DHPE 的运行时间。显然，DyHNE 比 DHPE 和 DANE 快

得多，这是因为 DHPE 的时间复杂度为  $O(T(f + g + N)d^2 + d^4)$ ，而 DANE 的为  $O(T(f + g + N)d^2)$ 。其中， $N$  是网络节点的总数。由于 DyHNE 使用元路径增强邻接矩阵对异质网络的变化进行建模，因此 DyHNE 模型只需要更新元路径集合中节点的表示，而无需以  $O(T(f + g + N_M)d^2)$  的复杂度更新所有节点的嵌入表示。

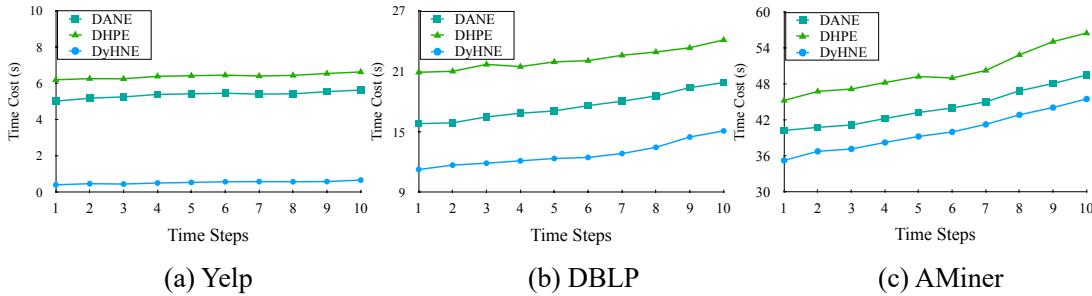


图 5-4 和对比方法相比 DyHNE 模型效率性

### 5.3.5.2 和静态 StHNE 模型相比

为了验证 DyHNE 相对于重新训练 StHNE（即 StHNE-retrain）的高效性，该实验在图 5-5 中比较了 DyHNE 相对 StHNE 的加速比。当嵌入表示大小约为 10 时，DyHNE 在 Yelp 上获得约 174 倍的加速比。在嵌入表示维度等于 100 时，DyHNE 的速度也比 StHNE 快 16 倍。总体而言，尽管加速比随节点向量维度的增加而降低，但 DyHNE 仍比重新训练 StHNE 高效得多。

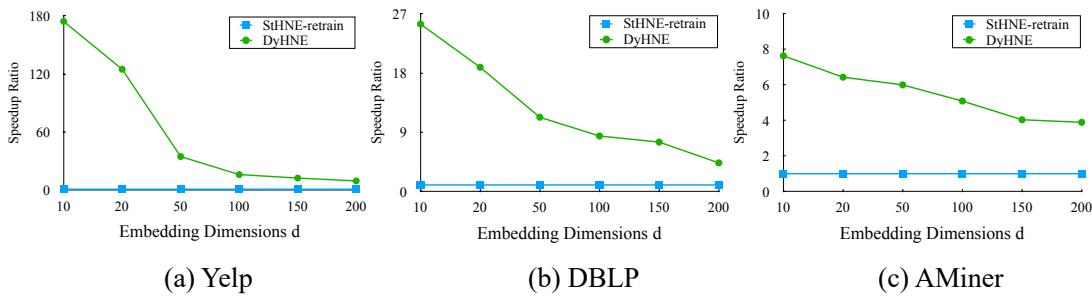


图 5-5 和 StHNE 相比 DyHNE 模型效率性

为了进一步探索 DyHNE 的效率，该实验计算模型的运行时间。图 5-6 展示了更新节点表示向量所需的时间随维数增加而逐渐增加，这与上文对时间复杂度的分析是一致的。

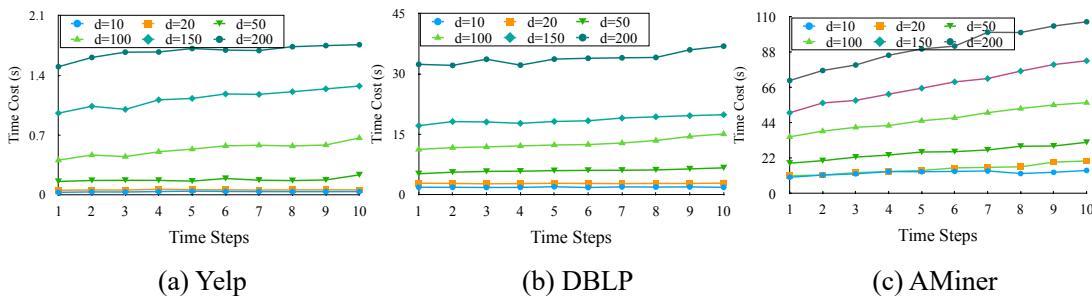


图 5-6 模型运行时间

### 5.3.6 参数分析实验

#### 5.3.6.1 元路径融合

由于 StHNE 和 DyHNE 对多个元路径进行了带权融合，因此该实验探索不同元路径对节点嵌入表示的影响。具体来说，本实验首先利用单个元路径学习节点的嵌入表示，然后均值同和元路径，最后采用最佳权重融合元路径。如图 5-7 所示，使用单个元路径很难完全对异质信息网络的拓扑结构和语义信息进行建模，从而导致分类性能下降。通过融合多个元路径，模型分类性能有所提升。但由于不同的元路径的拓扑结构和语义不同，因此将它们采用不同权重融合可获得最佳性能。

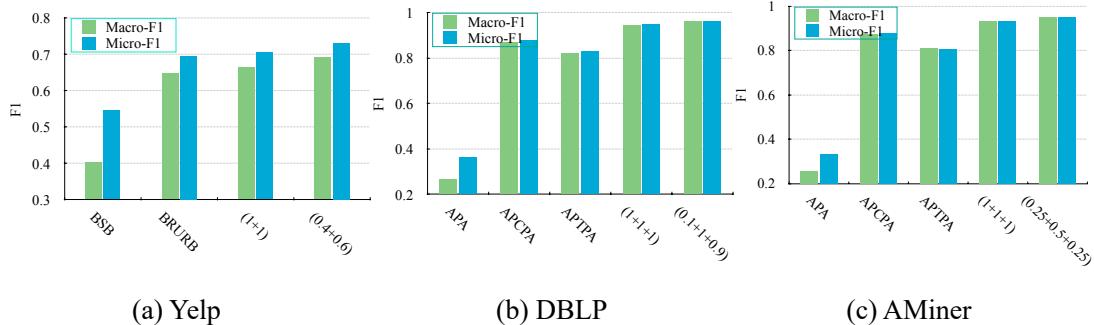


图 5-7 元路径的融合实验，(1+1) 表示分配给 BSB 和 BRURB 的权重分别为 1 和 1。

#### 5.3.6.2 表示向量维度

在该实验中，研究节点嵌入向量维度对节点分类任务的影响。具体来说，该实验将嵌入向量维度更改为 10、20、50、100、150 和 200。图 5-8 中报告了节点分类的结果。可以看到，随着维度的增加，模型的性能也会提高。当表示向量的维数达到 100 左右时，模型性能趋于稳定。这表明模型能够使用较低维度的向量表示捕获异质信息网络中各种关系的丰富信息。

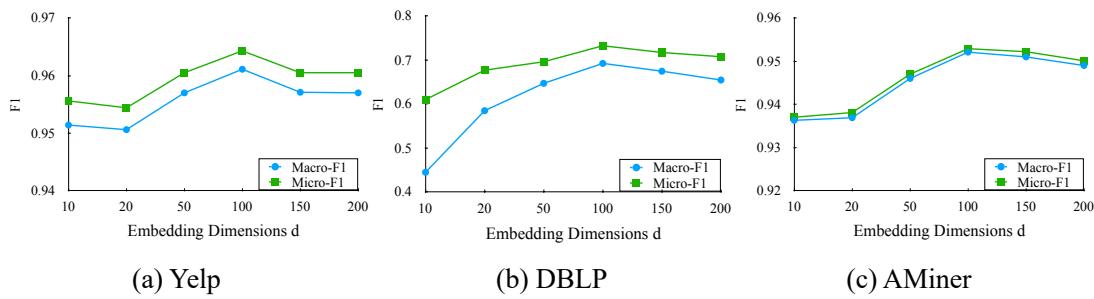


图 5-8 表示向量维度实验

## 5.4 本章小结

本章内容主要介绍了动态异质信息网络表示学习算法(DyHNE)。在 5.1 节，主要介绍了模型 DyHNE 的提出背景及创新贡献，并概述了模型的基本思想。5.2

节，详细地叙述了模型的设计与实现，首先 5.2.1 小节对相关符号和定义进行了介绍，其次 5.2.2 小节概述了模型的基本思想，然后 5.2.3 小节详细介绍了针对静态异质信息网络设计的基础模型 StHNE，在 5.2.4 小节从谱理论的角度对 StHNE 进行求解，最后在 5.2.5 小节叙述了针对动态异质信息网络嵌入表示更新的算法 DyHNE。在 5.3 节，主要报告了针对模型 StHNE 和 DyHNE 的实验和分析，包括 5.3.1 小节的实验设置，5.3.2 小节和 5.3.3 小节对模型有效性的验证实验，5.3.4 小节和 5.3.5 小节对模型效率性的分析，以及 5.3.6 小节对模型参数的分析实验。

## 第六章 用于好友增强推荐的社交影响力专注的神经网络

随着在线社交网络的兴起，许多社交应用程序中出现了一种新的推荐方式，在本文中称为“好友增强推荐(Friend-Enhanced Recommendation, FER)”。在 FER 中，对于某一从未交互过的商品，用户显式可见其好友对该商品的交互行为(这些好友被称作好友推荐圈)，例如微信“看一看”场景下的文章推荐场景。与传统社交推荐不同，FER 中特有的好友推荐圈可能会因为其重要性和可解释性而显着改变推荐结果。在本研究课题中，首次对微信“看一看”的文章推荐场景进行了形式化定义，即好友增强推荐场景，然后针对这一场景，提出了一种新颖的社交影响力专注的神经网络(SIAN)。SIAN 从异质信息网络的角度建模推荐场景，构建了注意力特征聚合器，其对多类型的异质信息进行节点级和类型级的特征聚合，并学习得到用户和商品的向量表示。更重要的是，SIAN 模型设计了一个社交影响力耦合器，以专心地捕捉好友推荐圈的影响力。实验结果表明，在两个真实数据集上，所提出的 SIAN 模型优于的最新的一些比较方法。

### 6.1 研究背景

随着在线社交网络的兴起，人们更愿意在社交平台上表达自己的观点并与好友共享信息。好友成为重要的信息来源和高质量的信息过滤器。好友互动过的商品(共享的，喜欢的等)会对用户产生重大影响，这很可能会成为用户未来的兴趣所在。许多推荐系统专注于好友的社交影响(例如，关注 YouTube 中的信息流和微信中的“看一看”文章推荐)。在本研究中，基于微信“看一看”文章推荐场景，形式化了一个全新的推荐场景，叫做好友增强的推荐系统(Friend-Enhanced Recommendation, FER)，其大大增强了推荐系统中好友对用户行为的影响力。相较于传统的社文化推荐，好友增强推荐有两个主要区别：(1) 鉴于好友可以看作高质量信息过滤器为用户提供高质量的商品，好友增强推荐只为用户推荐好友交互过的商品(读过的文章)；(2) 与某一商品交互过的所有好友都会显式的展示给当前用户，即当前用户已知哪些好友与当前物品有过交互。这一特点重点强调了显式的社交影响力对用户行为的影响，这也有益于用户行为的可解释性。

近些年，好友增强的推荐系统被广泛使用，例如微信的“看一看”文章推荐。图 6-1 展示了一个微信“看一看”场景下的形式化示例。对于每一个用户-物品对，好友增强的推荐显式地展示已经和当前物品交互过的好友集合，这个集合被定义为：针对当前商品，当前用户的好友推荐圈(Friend Referral Circle, FRC)。

举例来说，对于关于 AirPods 的文章，Jerry 的好友推荐圈是 {Tom, Lily, Jack}。这样的好友推荐圈极大地凸显了好友的社交影响力及其社交角色，这使得好友增强的推荐场景更加复杂，甚至改变了传统推荐过程。以图 6-1 为例，在传统社交推荐中，好友推荐圈对于 Jerry 是不可见的（不会展示给 Jerry），因此 Jerry 可能会基于自身的兴趣阅读某一篇文章。然而，在好友增强的推荐场景中，除了文章本身对用户的吸引以外，好友的影响可能成为用户点击文章的主要原因。在这里，好友推荐圈对用户 Jerry 是显式可见的，因此他阅读这篇关于 AirPods 的文章，更可能是因为科技专家 Tom 读过这篇文章。而 Jerry 阅读一篇关于 Disneyland 的文章，完全有可能是因为他的妻子 Lily 读过这篇文章。更进一步地，当文章和科技相关时，相较于妻子和科技文章，专家和文章的同时出现对 Jerry 的耦合影响力更大，而在娱乐方面，可能是完全相反的情况。因此，在好友增强的推荐场景中，多个因素导致了用户的行为。用户阅读/点击一篇文章的原因可能来自：(1) 其自身对文章的兴趣（文章本身）(2) 专家的推荐（文章-好友的组合）(3) 对某一好友的关注（好友）。在好友增强的推荐场景中，用户有窥探好友在看的内容的倾向，而非仅仅看自己本身感兴趣的内容。甚至可以说，社交推荐关注于结合社交信息去推荐物品，而好友增强的推荐旨在推荐物品和好友的组合。

作为好友增强推荐的关键特性，显式的好友推荐圈为该推荐场景带来两个挑战：(1) 如何从多方面的异质因素中提取关键信息？好友增强的推荐场景涉及多个异质目标，例如物品内容，好友推荐圈及用户-物品交互行为等。在用户，物品和朋友推荐圈的不同组合下，这些因素的影响甚至也有所不同。因此好友增强的推荐场景更具挑战性，因为它不仅需要学习用户对商品的偏好，还需要预测不同因素对好友的影响。(2) 如何利用显式的好友推荐圈信息？显式的好友推荐圈推荐极大地强调了推荐中社交信息的重要性，这在好友增强的推荐场景中是至关重要。但是，很少有工作在实际推荐中探索好友推荐圈的影响和特性。因此需要一种精心设计的策略，以充分利用好友增强的推荐场景中的显式的好友推荐圈信息。

为了解决这些问题，在本研究中提出了一种新颖的社交影响力专注的神经网络 (SIAN)。具体来说，SIAN 将好友增强推荐定义为异质社交图上的用户-物品交互预测任务，该任务将丰富的异质信息灵活地集成到异质对象及其交互连接中。首先，本研究特别设计了一个注意力特征聚合器，它同时考虑节点级和类型级的特征聚合，以学习用户和物品的表示向量。接下来，本研究实现了一个社交影响力耦合器，以建模通过显式好友推荐圈传播的耦合影响力，该耦合器利用注意力机制将多种因素（例如，朋友和物品）的影响力耦合在一起。总体而言，SIAN 模型捕获了好友增强的推荐场景中有价值的多方面因素，从而成功地从异质信息网络和显式好友推荐圈中提取了用户的最基本偏好。

综上所述，本研究课题的主要贡献可以总结为如下几点：

- 本研究首次探索了微信“看一看”中文章推荐场景，称作好友增强推荐，其中交互过的好友被显式附加到物品中，并展示给用户。
- 针对好友增强推荐场景，本研究提出了一种新颖的社交影响力专注的神经网络（SIAN）。它使用新颖的注意力特征聚合器来提取有用的多方面异质信息，并利用社交影响耦合器来判断不同朋友推荐的重要性。
- 在两个真实世界的数据集上的实验证明了 SIAN 模型的有效性和鲁棒性。进一步的定量分析也揭示了有价值的社会学模式，反映了当社交影响力变得更加重要时用户行为的变化和可解释性。

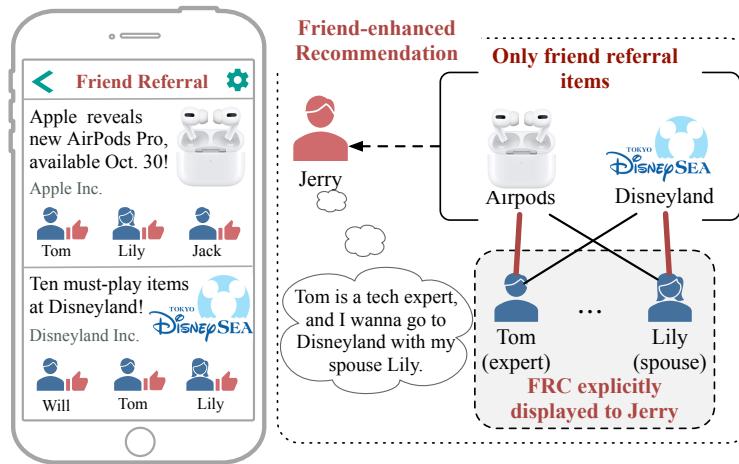


图 6-1 好友增强的推荐场景。左侧显示了推荐给 Jerry 的两篇文章的场景，下面是与两篇文章互动（共享，喜欢等）的朋友（例如 Tom）。右侧显示了好友增强推荐问题的形式化，其中仅推荐朋友交互过的文章，并且与该项目进行交互的朋友被明确暴露给用户 Jerry。

## 6.2 模型设计及实现

### 6.2.1 符号及定义

#### 定义 1：异质社交网络

异质社交网络定义为  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ ，其中  $\mathcal{V} = \mathcal{V}_U \cup \mathcal{V}_I$  和  $\mathcal{E} = \mathcal{E}_F \cup \mathcal{E}_R$  是节点和边的集合。 $\mathcal{V}_U$  和  $\mathcal{V}_I$  是用户和物品的集合。对于  $u, v \in \mathcal{V}_U$ ,  $\langle u, v \rangle \in \mathcal{E}_F$  表示用户之间的交友关系。对于  $u \in \mathcal{V}_U$  和  $i \in \mathcal{V}_I$ ,  $\langle u, i \rangle \in \mathcal{E}_R$  是  $u$  和  $i$  之间的交互关系。通过添加属性特征或边关系，可以将异质设计网络进一步扩展为异质信息网络。

#### 定义 2：好友推荐圈（Friend Referral Circle, FRC）

给定一个异质社交网络  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ ，对于一个从未交互过的物品  $i$ （即  $\langle u, i \rangle \notin \mathcal{E}_R$ ），用户  $u$  的好友推荐圈可以定义为  $\mathcal{C}_u(i) = \{v | \langle u, v \rangle \in \mathcal{E}_F \cap \langle v, i \rangle \in \mathcal{E}_R\}$ ，这里  $v$  被称作用于用户  $u$  的有影响力的好友。

### 定义 3：好友增强推荐（Friend-Enhanced Recommendation, FER）

给定异质社交图  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$  和用户  $u$  针对未交互物品  $i$  的好友推荐圈  $\mathcal{C}_u(i)$ ，好友增强的推荐旨在预测用户  $u$  是否对物品  $i$  有潜在的偏好。也就是说，学习预测函数  $\hat{y}_{ui} = \mathcal{F}(\mathcal{G}, \mathcal{C}_u(i); \theta)$ ，其中  $\hat{y}_{ui}$  是用户  $u$  和物品  $i$  交互的可能性， $\theta$  是模型参数。

#### 6.2.2 模型总览

如图 6-2 所示，SIAN 利用异质社交网络建模好友增强的推荐场景。除了用户和物品的向量表示（例如，例如， $\mathbf{h}_u$  表示用户 Jerry 的嵌入向量， $\mathbf{h}_i$  是 Disneyland 相关的文章的表示向量），SIAN 还通过耦合有影响力的好友与物品来学习社交影响力的低维向量表示（即  $\mathbf{h}_{ui}$ ）。通过学习用户、物品和耦合的社交影响力的向量表示，最终 SIAN 预测用户  $u$  和物品  $i$  之间交互的概率  $\hat{y}_{ui}$ 。具体来说，对于异质社交网络中的每个节点都装备带有节点级和类型级的注意力特征聚合器。在每个级别，SIAN 设计一种注意力机制来区分和捕获邻居和类型的潜在关联性，这中分层的注意力机制使得 SIAN 模型能够更细粒度地编码多方面的异质信息。与之前的一些工作<sup>[37,53,57]</sup>不同的是，SIAN 不需要任何先验知识，用于手动选择元路径。其次，在 SIAN 中，设计了一个社交影响力耦合器，其用来捕获一个有影响力好友（例如 Tom）和一个物品（例如 Disneyland 相关的文章）的耦合影响力，从而量化它们的耦合影响力程度。最后，带有注意力地融合来自好友推荐圈内的多个耦合影响力，以表示整个好友推荐圈对当前用户和物品的影响。

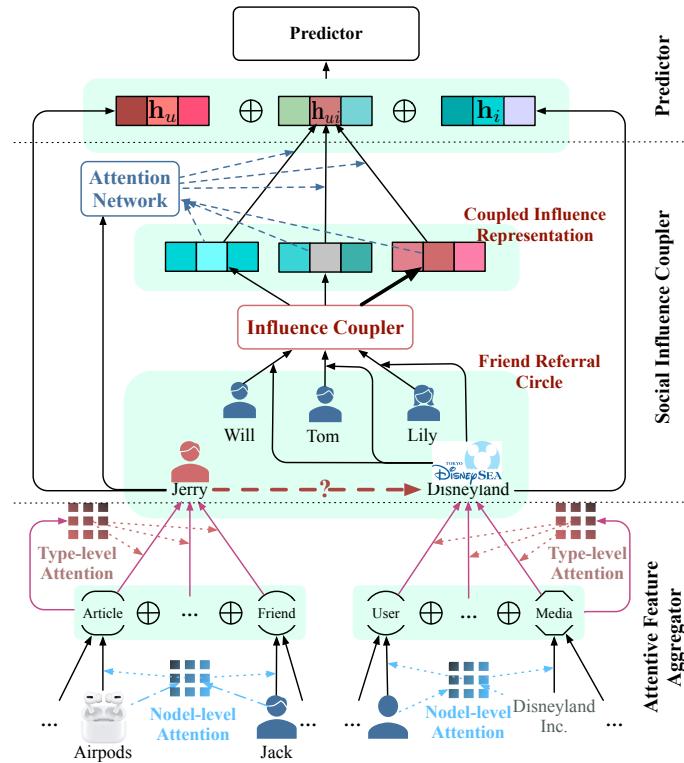


图 6-2 模型 SIAN 总览

### 6.2.3 注意力特征聚合器

注意力特征聚合器旨在学习用户和物品的嵌入表示。考虑到相同类型的不同邻居可能对特征聚合的贡献是不同的，并且不同类型包含多方面的信息，因此本研究设计了分层的节点级和类型级的注意力特征聚合器。节点级聚合以细粒度的方式分别对用户/物品特征进行聚合建模，而类型级聚合则捕获异质的类型信息。

#### 6.2.3.1 节点级注意力聚合

给定用户  $u$ ，定义  $\mathcal{N}_u = \mathcal{N}_u^{t_1} \cup \mathcal{N}_u^{t_2} \cup \dots \cup \mathcal{N}_u^{t_{|\mathcal{T}|}}$  为邻居集合，其是  $|\mathcal{T}|$  个类型邻居集合的并集。对于  $t \in \mathcal{T}$  类型邻居，在  $t$  类型空间的特征聚合定义为如下函数：

$$\mathbf{p}_u^t = \text{ReLU} \left( \mathbf{W}_p \left( \sum_{k \in \mathcal{N}_u^t} \alpha_{ku} \mathbf{x}_k \right) + \mathbf{b}_p \right) \quad (6-1)$$

其中  $\mathbf{p}_u^t \in \mathbb{R}^d$  是用户  $u$  在  $t$  类型空间聚合后的隐含表示， $\mathbf{x}_k \in \mathbb{R}^d$  是邻居节点  $k$  的初始表示，其可以是随机初始化或由给定特征初始化。这里的  $\mathbf{W}_p \in \mathbb{R}^{d \times d}$  和  $\mathbf{b}_p \in \mathbb{R}^d$  是神经网络的权重和偏置向量。 $\alpha_{ku}$  是邻居节点  $k$  对于  $u$  的特征聚合的贡献：

$$\alpha_{ku} = \frac{\exp(f([\mathbf{x}_k \oplus \mathbf{x}_u]))}{\sum_{k' \in \mathcal{N}_u^t} \exp(f([\mathbf{x}_{k'} \oplus \mathbf{x}_u]))} \quad (6-2)$$

这里  $f(\cdot)$  是一个由 ReLU 激活的两层神经网络， $\oplus$  表示向量拼接操作。显然地， $\alpha_{ku}$  数值越大，邻居节点  $k$  对用户  $u$  的特征聚合的贡献越大。

给定多种类型的邻居，可以得到在不同类型空间内邻居的聚合特征，即一个节点表示的集合： $\{\mathbf{p}_u^{t_1}, \dots, \mathbf{p}_u^{t_{|\mathcal{T}|}}\}$ 。

#### 6.2.3.2 类型级注意力聚合

由于每个节点多方面信息有不同的偏好，给定一个用户  $u$  以及其在不同类型空间的节点级特征聚合表示，SIAN 学习不同类型的权重并聚合不同类型的特征：

$$\mathbf{h}_u = \text{ReLU} \left( \mathbf{W}_h \sum_{t \in \mathcal{T}} \beta_{tu} \mathbf{p}_u^t + \mathbf{b}_h \right) \quad (6-3)$$

其中  $\mathbf{h}_u \in \mathbb{R}^d$  是  $u$  的隐含表示。 $\{\mathbf{W}_h \in \mathbb{R}^{d \times d}, \mathbf{b}_h \in \mathbb{R}^d\}$  是模型参数。由于多类型的邻居包含多方面信息并且这些信息是互相关联的， $\beta_{tu}$  表示类型  $t$  对于节点  $u$  的特征聚合的权重。对于  $u$ ，通过拼接邻居特征聚合的向量表示，定义如下的权重：

$$\beta_{tu} = \frac{\exp(\mathbf{a}_t^\top [\mathbf{p}_u^{t_1} \oplus \mathbf{p}_u^{t_2} \oplus \dots \oplus \mathbf{p}_u^{t_{|\mathcal{T}|}}])}{\sum_{t' \in \mathcal{T}} \exp(\mathbf{a}_{t'}^\top [\mathbf{p}_u^{t_1} \oplus \mathbf{p}_u^{t_2} \oplus \dots \oplus \mathbf{p}_u^{t_{|\mathcal{T}|}}])} \quad (6-4)$$

其中  $\mathbf{a}_t \in \mathbb{R}^{Td}$  是共享于所有用户的类型感知向量。基于上式，多种类型邻居特征的聚合可以为用户捕获丰富的多方面信息，并且  $\mathbf{a}_t$  编码了每个类型的全局偏好性。同样地，对于每个物品  $i$ ，可以通过注意力特征聚合器得到其隐含表示  $\mathbf{h}_i$ 。

## 6.2.4 社交影响力耦合器

### 6.2.4.1 耦合的影响力表示

如<sup>[89,90]</sup>所研究的，人类的行为受到多种因素的影响。在好友增强的推荐场景中，用户 $u$ 是否与物品 $i$ 进行交互不仅有物品本身或用户好友决定，好友和物品的同时出现可能会产生更大的影响。正如前文示例（图 6-1），当 Jerry 面对与科技技术相关的文章时，专家（例如 Tom）和物品（例如 AirPods 相关的文章）之间的耦合比配偶和科技文章之间的耦合具有更大的影响力，但与娱乐相关的文章可能会发生相反的情况。因此，给定用户 $u$ ，物品 $i$ 和好友推荐圈 $\mathcal{C}_u(i)$ ，可以将每个好友 $v \in \mathcal{C}_u(i)$ 和物品 $i$ 的影响力耦合为：

$$\mathbf{c}_{\langle v, i \rangle} = \sigma(\mathbf{W}_c \phi(\mathbf{h}_v, \mathbf{h}_i) + \mathbf{b}_c) \quad (6-5)$$

其中 $\mathbf{h}_v$ 和 $\mathbf{h}_i$ 是用户 $v$ 和物品 $i$ 特征聚合后的向量表示。 $\phi(\cdot, \cdot)$ 表示一个融合函数，可以是逐元素的相乘、相加或者拼接（这里采用拼接）。显然地，公式(6-5)耦合了物品 $i$ 和有影响力的好友 $v$ 的特征，因此有能力捕获两者的影响力。

### 6.2.4.2 影响力强度

基于耦合的影响力表示 $\mathbf{c}_{\langle v, i \rangle}$ ，接下来的目标是获得耦合影响力对用户 $u$ 的影响强度。由于影响强度取决于用户 $u$ ，因此将用户的表示向量 $\mathbf{h}_u$ 整合到影响力中：

$$d'_{u \leftarrow \langle v, i \rangle} = \sigma(\mathbf{W}_2 (\sigma(\mathbf{W}_1 \phi(\mathbf{c}_{v, i}, \mathbf{h}_u) + \mathbf{b}_1)) + b_2) \quad (6-6)$$

其中 $\{\mathbf{W}_1, \mathbf{W}_2, \mathbf{b}_1, b_2\}$ 是一个两层神经网络的参数。接下来，影响力强度可以通过归一化上述影响力得到，其表示了在推荐物品 $i$ 时，好友 $v$ 对用户 $u$ 行为的影响力：

$$d_{u \leftarrow \langle v, i \rangle} = \frac{\exp(d'_{u \leftarrow \langle v, i \rangle})}{\sum_{v' \in \mathcal{C}_u(i)} \exp(d'_{u \leftarrow \langle v', i \rangle})} \quad (6-7)$$

由于好友的影响力从好友推荐圈中传播而来，SIAN 模型进一步提出融合好友推荐圈内所有好友对用户 $u$ 的影响：

$$\mathbf{h}_{ui} = \sum_{v \in \mathcal{C}_u(i)} d_{u \leftarrow \langle v, i \rangle} \mathbf{c}_{\langle v, i \rangle} \quad (6-8)$$

由于耦合影响力表示 $\mathbf{c}_{\langle v, i \rangle}$ 融合有影响力好友以及物品的隐含因素，公式(6-8)保证了隐含表示 $\mathbf{h}_{ui}$ 可以有效地编码在好友推荐圈内传播的社交影响力。

## 6.2.5 行为预测及模型训练

基于用户、物品以及耦合的影响力表示，将三者拼接并送入两层神经网络中：

$$\mathbf{h}_o = \sigma(\mathbf{W}_{o_2} (\sigma(\mathbf{W}_{o_1} ([\mathbf{h}_u \oplus \mathbf{h}_{ui} \oplus \mathbf{h}_i]) + \mathbf{b}_{o_1}) + \mathbf{b}_{o_2}) \quad (6-9)$$

接下来，通过一个回归层，可以预测得到用户-物品的交互概率：

$$\hat{y}_{ui} = \text{sigmoid}(\mathbf{w}_y^\top \mathbf{h}_o + b_y) \quad (6-10)$$

最终，通过优化下面的交叉熵损失函数训练优化模型：

$$-\sum_{\langle u,i \rangle \in \mathcal{E}_R} (y_{ui} \log \hat{y}_{ui} + (1 - y_{ui}) \log (1 - \hat{y}_{ui})) + \lambda \|\Theta\|_2^2 \quad (6-11)$$

其中  $y_{ui}$  是真实交互标签,  $\lambda$  是用于避免过拟合的 L2 正则化参数。

## 6.3 实验分析

### 6.3.1 实验设置

#### 6.3.1.1 数据集

本实验在两个数据集上评估模型, 表 6-1 汇总了数据集的统计信息。

- Yelp 是既包含互动又包含社交关系的电商评论数据集。在本实验中, 首先随机采样一组用户, 对于每个用户  $u$ , 根据给定的用户-用户好友关系和用户-物品交互关系构造一组好友推荐圈  $\{\mathcal{C}_u(i) : i \in \mathcal{V}_I\}$ , 其中对于好友推荐圈为空集合的交互数据被过滤掉。为了节点的初始特征向量, 本实验利用 word2vec<sup>[23]</sup> 学习评论文本的向量表示作为节点的初始向量。
- 朋友在看数据 (FWD) 抽取自微信“看一看”文章推荐场景。在该数据中, 每个用户或物品 (即文章) 关联一些特征 (例如年龄或文本内容向量)。

表 6-1 数据集统计信息

Datasets	Nodes	#Nodes	Relations	#Relations
Yelp	User (U)	8,163	U-U	92,248
	Item (I)	7,900	U-I	36,571
FWD	User (U)	72,371	U-U	8,639,884
	Article (A)	22,218	U-A	2,465,675
	Media (M)	218,887	U-M	1,368,868
			A-M	22,218

#### 6.3.1.2 对比方法

在该实验中, 将 SIAN 模型与四类代表性的方法进行了比较, 包括: (1) 基于特征/结构的方法, 即 MLP<sup>[91]</sup>, DeepWalk (简写作 dw)<sup>[24]</sup>, node2vec<sup>[25]</sup>, 和 metapath2vec (简写作 mp2vec)<sup>[37]</sup>。(2) 基于特征或结构融合的方法, 即 dw+fea, node2vec+fea 和 mp2vec+fea。在该实验中, 将模型学习到的节点嵌入表示和原始特征拼接, 并将其送入逻辑回归以预测用户-物品交互概率。(3) 图神经网络方法, 即 GCN<sup>[27]</sup>, GAT<sup>[28]</sup>和 HAN<sup>[57]</sup>。在该实验中利用这些模型学习节点的向量表示, 然后预测用户-物品交互概率。(4) 社会化推荐方法, 即 TrustMF<sup>[92]</sup>和 DiffNet<sup>[93]</sup>, 在该实验中采用论文建议的模型结构和参数预测用户-物品的交互概率。

#### 6.3.1.3 参数设置

在该实验中, 对于每个数据集, 训练, 验证和测试集的比例为 7: 1: 2。本论文所提出的模型 SIAN 利用 PyTorch 实现, 并采用 Adam 优化方法进行梯度下

降优化。对于模型中超参数，根据验证集上的 AUC 指标，该实验使用网格搜索将学习率，批处理大小和正则化参数设置为 0.001、1、024 和 0.0005。对于对比方法，该实验参看其原始论文优化模型参数。最后，对于所有方法，该实验都会报告它们在不同向量维度 32 和 64 维下的性能。

### 6.3.2 实验结果及分析

在该实验中，采用三个广泛使用的度量标准 AUC，F1 和 Accuracy 来评估用户-物品交互预测的性能。表 6-2 报告了在不同维度的节点嵌入表示下的模型性能，从中可以得到以下一些发现。

表 6-2 行为预测实验结果

Model	Yelp						FWD					
	AUC		F1		Accuracy		AUC		F1		Accuracy	
	d=32	d=64										
MLP	0.6704	0.6876	0.6001	0.6209	0.6589	0.6795	0.5094	0.5182	0.1883	0.1932	0.2205	0.2302
dw	0.7693	0.7964	0.6024	0.6393	0.7001	0.7264	0.5587	0.5636	0.2673	0.2781	0.1997	0.2056
node2vec	0.7903	0.8026	0.6287	0.6531	0.7102	0.7342	0.5632	0.5712	0.2674	0.2715	0.2699	0.2767
mp2vec	0.8194	0.8346	0.6309	0.6539	0.7076	0.7399	0.5744	0.5834	0.2651	0.2724	0.4152	0.4244
dw+fea	0.7899	0.8067	0.6096	0.6391	0.7493	0.7629	0.5301	0.5433	0.2689	0.2799	0.2377	0.2495
node2vec+fea	0.8011	0.8116	0.6634	0.6871	0.7215	0.7442	0.5672	0.5715	0.2691	0.2744	0.3547	0.3603
mp2vec+fea	0.8301	0.8427	0.6621	0.6804	0.7611	0.7856	0.5685	0.5871	0.2511	0.2635	0.4698	0.4935
GCN	0.8022	0.8251	0.6779	0.6922	0.7602	0.7882	0.5875	0.5986	0.2607	0.2789	0.4782	0.4853
GAT	0.8076	0.8456	0.6735	0.6945	0.7783	0.7934	0.5944	0.6006	0.2867	0.2912	0.4812	0.4936
HAN	0.8218	0.8476	0.7003	0.7312	0.7893	0.8102	0.5913	0.6025	0.2932	0.3011	0.4807	0.4937
TrustMF	0.8183	0.8301	0.6823	0.7093	0.7931	0.8027	0.6001	0.6023	0.3013	0.3154	0.5298	0.5404
DiffNet	0.8793	0.8929	0.8724	0.8923	0.8698	0.8905	0.6418	0.6594	0.3228	0.3379	0.6493	0.6576
SIAN	<b>0.9486</b>	<b>0.9571</b>	<b>0.8976</b>	<b>0.9128</b>	<b>0.9096</b>	<b>0.9295</b>	<b>0.6845</b>	<b>0.6928</b>	<b>0.3517</b>	<b>0.3651</b>	<b>0.6933</b>	<b>0.7018</b>

(1) 在配对  $t$  检验下 ( $p < 0.01$ )，在两个数据集上，针对不同评价指标，SIAN 模型的表现均优于所有对比方法。这表明在好友增强推荐场景下的多方面因素中，模型 SIAN 可以很好地建模用户的核心兴趣点。这些性能提升既得益于从节点级和类型级的注意力特征聚合生成的高质量节点表示，也得益于本研究所提出的社交影响力耦合器，它可以挖掘出用户对社交好友的偏好。此外，预测效果在不同维度上的一致提升证明了 SIAN 对于节点的表示维度具有较好的鲁棒性。

(2) 与图神经网络方法相比，SIAN 模型的显著提升证明了节点级和类型级注意力特征聚合的有效性。特别是，SIAN 的性能优于 HAN，后者也是为具有两级聚合的异质信息网络而设计的。这是因为 SIAN 中的类型级别关注多个方面聚合捕获的异质信息，而不受 HAN 中使用的预定义元路径的限制。而且，这些改进也表明了 SIAN 中社交影响力耦合器在好友增强推荐中的重要性。

(3) 社会化推荐的对比方法也取得了较好的表现，这进一步证实了社交影响力在好友增强推荐中的重要性。与仅将社交关系视为辅助信息的其他基于 GNN 的模型相比，SIAN 模型的显著提升意味着好友推荐圈可能在好友增强推荐

中占据主导地位，应谨慎建模。特别是，SIAN 取得了最佳性能，也再次证明了社交影响力耦合器在为好友增强推荐场景编码各种社交因素时的能力。

### 6.3.3 多方面信息影响分析

在注意力特征聚合器中，每个节点的向量表示都是从其具有不同权重的各种类型的异质邻居聚合而来的。在该实验中，通过计算所有实例之间的平均类型级别的注意力权重来分析异质因素对用户嵌入表示的贡献。

如图 6-3 所示，两个数据集上好友类型的平均注意力权重值显著大于物品类型的平均注意力权重。令人惊讶的是，该模型更加关注用户的社交关系，与传统推荐相比有明显的不同，在传统推荐中，用户与物品之间的交互关系被认为是更关键的信息。这也证明了 SIAN 中构建的社交影响力耦合器的合理性，该耦合器在从好友推荐圈中建模用户偏好方面起着重要作用。

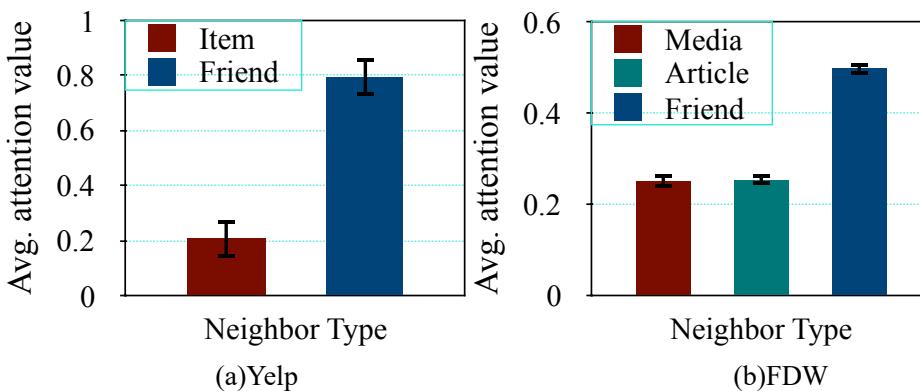


图 6-3 基于注意力权重的多方面信息影响分析

### 6.3.4 社交影响力分析

以上实验已经验证了好友推荐圈是好友增强推荐中最重要的因素。但是，朋友可能会从不同方面（例如权威度，亲密性或兴趣相似度）影响用户行为。接下来，该实验将展示不同的用户属性如何通过社交影响力耦合器的影响力强度来影响好友增强推荐场景中的用户行为。由于在 FWD 数据中具有详细的用户属性信息，因此本实验在该数据集上进行分析。

#### 6.3.4.1 评估设定

社交影响力耦合器中的影响力强度反映了不同朋友的重要性。在为用户  $u$  推荐物品  $i$  的情况下，假设具有最高影响力的朋友  $v$ （即  $d_{u \leftarrow (v,i)}$ ）是最有影响力的朋友，并且  $v$  的所有属性信息值对其影响力的贡献都是相等的。给定用户属性信息和一个用户组，该实验通过计算该组用户的好友推荐圈中所有好友的属性值来定义背景分布（background distribution），即在真实推荐场景下的曝光分布。而通过

计算用户组中影响力强度最大的朋友的属性值来定义影响分布 (influence distribution)，即 SIAN 的预测分布情况。因此，背景分布代表该用户组中好友的普遍特征，而影响分布则代表用户组内最有影响力朋友的特征。如果两个分布完全一致，那么此属性不是影响用户组的关键社交因素。反之，两个分布之间的差异意味着该属性在多大程度上是关键的社交因素，以及其差异如何影响用户行为。

### 6.3.4.2 结果分析

在图 6-4 中，报告了在四个不同属性下（即权威度、年龄、性别和地理位置）的两个分布，并有以下发现。

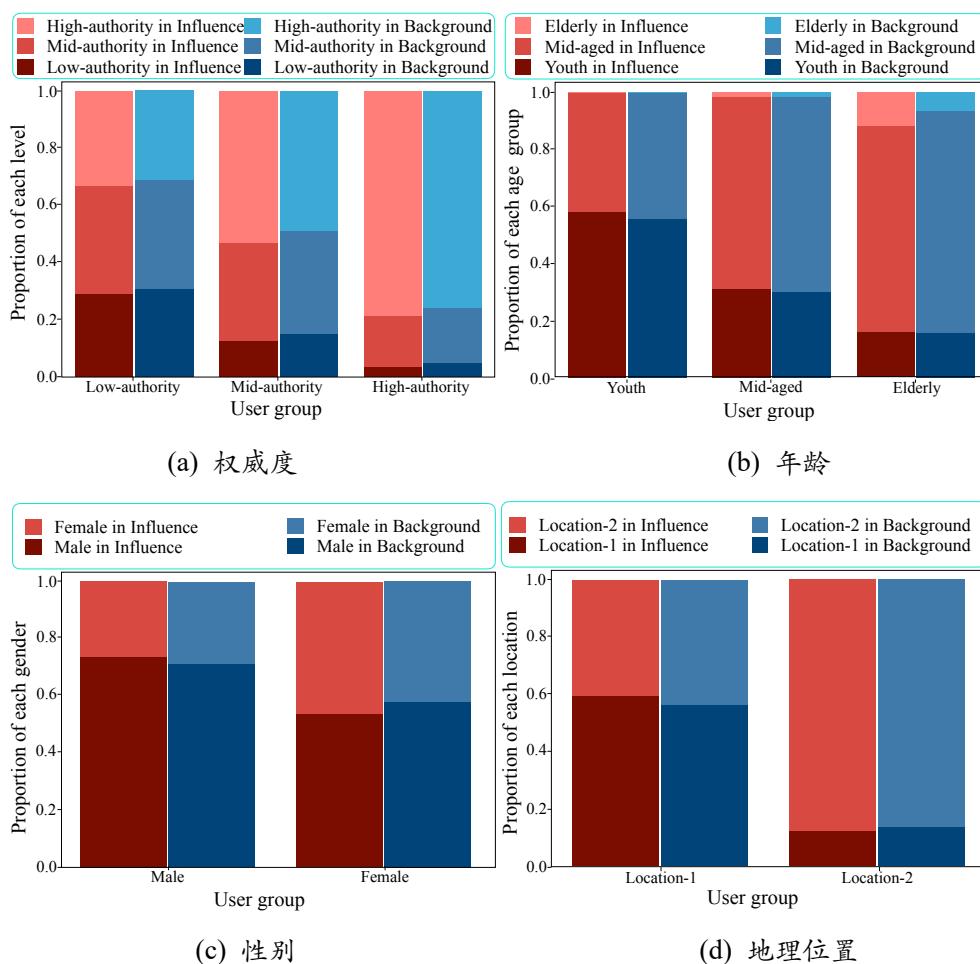


图 6-4 基于用户属性的社交影响力分析。对于每个属性和用户组（例如(a)中的权限属性和低权限用户组），左图表示影响分布，而右图表示背景分布。在每个栏中，每个不同颜色段的高度指示属性值在影响或背景分布中的比例。

(1) 在图 6-4(a) 中，可以观察到用户行为受到更具权威性的朋友的影响，而与用户自己的权威度无关。在权威度各不相同的所有三个用户组中，影响分布中的高权威比例大于背景分布中的高权威比例。例如，在中等权威度用户组中，顶部的红色块（高权威度影响分布）大于顶部的蓝色块（高权威度背景分布），这意味着高权威朋友对中等权威用户具有更大的影响力。结果并不令人惊讶，因为用户通常更容易受到权威人士的影响，这与常识相符。该实验结果还揭示了好

友增强推荐场景中的一个有趣现象，即有时用户会更多地关注老板或某一权威人士的喜好，而不是他们自己的实际喜好。

(2) 该实验还对其他用户属性进行了分析。可以发现，用户很容易受到与自己相似的朋友的影响。具体来说，图 6-4(b)表明人们喜欢同龄人推荐的物品（即微信文章），特别是对于年轻人和老年人。同时，图 6-4(c)和(d)显示用户倾向于观看具有相同性别或位置的朋友推荐的文章。在协同过滤中广泛采用的基于用户相似性的推荐，即使在好友增强的推荐场景中也仍然是存在的。

总而言之，尽管不同的社交因素对目标用户有各种影响，但没有一个因素占主导地位，这进一步显示了好友增强推荐场景的复杂性。在这种情况下，SIAN 对用户-物品交互预测的显著提升表明它可以很好地捕捉好友增强场景中的多方面社交因素，这可能有助于理解可解释的推荐。

## 6.4 本章小结

本章内容介绍了用于好友增强推荐的社交影响力专注的神经网络 (SIAN)。在 6.1 节，介绍了 SIAN 的提出背景及创新贡献。6.2 节，详细地叙述了模型的设计与实现，首先 6.2.1 小节对相关符号和定义进行了介绍，其次 6.2.2 小节概述了模型的基本架构，然后 6.2.3 小节详细介绍了注意力特征聚合器，在 6.2.4 小节介绍了社交影响力耦合器，最后在 6.2.5 小节叙述了 SIAN 的优化模型和训练方式。在 6.3 节，主要报告了针对 SIAN 的实验和分析，包括 6.3.1 小节的实验设置，6.3.2 小节的有效性实验，6.3.3 小节对异质社交网络中多方面信息影响的分析，以及 6.3.4 小节对社交影响力分析实验。

## 第七章 总结与展望

### 7.1 总结

在真实世界中，网络数据往往是动态变化、节点类型和边类型多种多样的，即所谓的动态信息网络和异质信息网络，这使得传统方法学习无法有效建模网络的真实特性信息，学到的节点向量表示很难适用于真实应用场景。因此，本文立足于研究动态、异质的信息网络表示学习算法，建模信息网络的本质特性，并在真实应用场景中验证有效性。具体而言，本文首先研究了同质信息网络的动态性问题，并提出结合微观和宏观动态性的动态网络表示学习算法 M<sup>2</sup>DNE。其次，本文研究了异质信息网络的异质性问题，并提出关系结构感知的异质信息网络表示学习算法 RHINE。然后，本文研究了信息网络的动态性和异质性问题，并提出了动态异质信息网络表示学习算法 DyHNE。最后，本文在微信“看一看”的文章推荐场景下，研究了好友增强的推荐问题，并提出社交影响力专注的神经网络 SIAN，以验证异质信息网络表示学习在真实应用场景下的有效性。

首先，为了研究同质信息网络的动态性问题，本论文提出了结合微观和宏观动态性的动态网络表示学习算法 M<sup>2</sup>DNE。具体地，对于微观动态性，M<sup>2</sup>DNE 将网络中边的建立视作一系列按时间顺序排列的事件的发生，并利用了一个时序注意力点过程来捕获网络结构的细致形成过程。对于宏观动态性，M<sup>2</sup>DNE 定义了一个通用的动力学方程，由网络嵌入表示参数化。该动力学方程用以捕获网络内部的演变模式，并在更高阶的结构层次对网络表示施以约束。时序动态网络中，微观动态性和宏观动态性的交互的演变交替地影响节点表示的学习过程。在三个真实的时序动态网络上，丰富的实验验证了 M<sup>2</sup>DNE 显著地优于最新的一些方法。

其次，为了研究异质信息网络的异质性问题，本论文提出了关系结构感知的异质信息网络表示学习模型 RHINE。通过透彻地对四个真实异质信息网络数据的数学分析和探索，本论文提出了两种结构相关的度量方法，这两种方法可以将异质的多类型关系一致地划分为两类：隶属关系（Affiliation Relations, ARs）和交互关系（Interaction Relations, IRs）。为了充分建模关系中有区别的特征，本论文提出两种特别设计的模型分别处理 ARs 和 IRs，以此可以保证捕获网络的拓扑结构和语义信息。最后，联合各个子模型，并以统一的方式训练优化 RHINE 模型。在四个真实数据集上的丰富实验证明了模型 RHINE 的有效性。在多个数据挖掘任务中，包括聚类、分类、链路预测等，本论文提出的模型 RHINE 的表现均优于现有的网络表示学习模型。

然后,为了研究信息网络的动态性和异质性,本论文提出了动态异质信息网络表示学习算法 DyHNE。更具体地说,本论文首先设计了一个基础模型 StHNE,用于保留网络中的结构和语义信息。基于谱理论,该模型通过解决广泛特征值问题,学习节点的表示向量。随着异质信息网络的演化,利用元路径增强的邻接矩阵的摄动,捕获网络的变化。然后基于特征值摄动理论,模型可以高效地更新节点的表示向量。在三个真实的异质信息网络上进行了丰富的实验,实验结果表明所提模型在有效性和效率性上均优于最新的表示学习方法。

最后,为了验证异质信息网络在真实应用场景中的有效性,本文进一步在微信“看一看”的文章推荐场景下,研究了好友增强的推荐问题,并提出社交影响力专注的神经网络 SIAN。为了融合丰富的异质信息,SIAN 从异质社交网络的角度建模推荐场景,构建了注意力特征聚合器,其对多类型的异质信息进行节点级和类型级的特征聚合,并学习得到用户和商品的向量表示。更重要的是,SIAN 模型设计了一个社交影响力耦合器,以专心地捕捉好友推荐圈的影响力。实验结果表明,在两个真实数据集上,所提出的 SIAN 模型优于的最新的一些比较方法。

## 7.2 展望

基于本文研究的四个研究内容,下一步研究方向可以总结为如下几个方面:

1. 在动态同质信息网络表示学习方面,本文提出的算法 M<sup>2</sup>DNE 采用时序点过程和动力学方程建模网络的微观与宏观动态性,其忽略了网络的一些突发演变情况,例如网络急剧增长或衰减等。因此,下一步工作可以尝试考虑网络动态演变过程中的突发情况,从而更好的学习动态网络中的节点表示。
2. 在异质信息网络表示学习方面,本文提出的模型 RHINE 通过两个指标区分网络中的多类型关系,进而对其区分建模。这是一种两阶段的做法,与拓扑结构相关的指标没有编码在模型构中,因此下一步工作可以尝试将结构性区分指标融入模型设计中,构建端到端的异质信息网络表示学习模型。
3. 在异质信息网络的动态性建模方面,本文提出的 DyHNE 聚焦于快照动态网络的增量表示学习,而没有考虑动态异质信息网络中的连续时序动态性。因此下一步工作可以尝试建模动态异质信息网络更细粒度的演变模式。
4. 在异质信息网络表示学习应用方面,本文研究了微信“看一看”场景下的文章推荐场景,并对该场景进行了简化处理,即没有考虑动态的推荐场景。因此下一步工作可以尝试结合异质信息网络表示学习和在线推荐场景,进一步验证网络表示学习在真实场景下的有效性。

## 参考文献

- [1] Cai H, Zheng V W, Chang K C-C. A comprehensive survey of graph embedding: Problems, techniques, and applications[J]. IEEE Transactions on Knowledge and Data Engineering, 2018, 30(9): 1616-1637.
- [2] Scott J. Social network analysis[J]. Sociology, 1988, 22(1): 109-127.
- [3] 吴信东, 李毅, 李磊. 在线社交网络影响力分析[J]. 计算机学报, 2014, 37(4): 735-752.
- [4] Shi C, Li Y, Zhang J, et al. A survey of heterogeneous information network analysis[J]. IEEE Transactions on Knowledge and Data Engineering, 2016, 29(1): 17-37.
- [5] Sun Y, Han J. Mining heterogeneous information networks: principles and methodologies[J]. Synthesis Lectures on Data Mining and Knowledge Discovery, 2012, 3(2): 1-159.
- [6] 黄立威, 李德毅, 马于涛, et al. 一种基于元路径的异质信息网络链路预测模型[J]. 计算机学报, 2014, 37(4): 848-858.
- [7] 汪小帆, 李翔, 陈关荣. 复杂网络理论及其应用[M]. 清华大学出版社有限公司, 2006.
- [8] Strogatz S H. Exploring complex networks[J]. nature, 2001, 410(6825): 268-276.
- [9] Boccaletti S, Latora V, Moreno Y, et al. Complex networks: Structure and dynamics[J]. Physics reports, 2006, 424(4-5): 175-308.
- [10] Barrat A, Barthelemy M, Vespignani A. Dynamical processes on complex networks[M]. Cambridge university press, 2008.
- [11] Carley K M. Dynamic network analysis[M]. Committee on Human Factors, 2003.
- [12] Cvetkovic D M, Doob M, Sachs H. Spectra of graphs[M]. 10. Academic Press, New York, 1980.
- [13] Sainath T N, Kingsbury B, Sindhwan V, et al. Low-rank matrix factorization for deep neural network training with high-dimensional output targets[C]. 2013 IEEE international conference on acoustics, speech and signal processing, 2013: 6655-6659.
- [14] Mnih A, Salakhutdinov R R. Probabilistic matrix factorization[C]. Advances in neural information processing systems, 2008: 1257-1264.
- [15] Lee D D, Seung H S. Algorithms for non-negative matrix factorization[C]. Advances in neural information processing systems, 2001: 556-562.

- [16] Bengio Y, Courville A, Vincent P. Representation learning: A review and new perspectives[J]. IEEE transactions on pattern analysis and machine intelligence, 2013, 35(8): 1798-1828.
- [17] Hamilton W, Ying Z, Leskovec J. Inductive representation learning on large graphs[C]. Advances in neural information processing systems, 2017: 1024-1034.
- [18] Cui P, Wang X, Pei J, et al. A survey on network embedding[J]. IEEE Transactions on Knowledge and Data Engineering, 2018, 31(5): 833-852.
- [19] Zhang D, Yin J, Zhu X, et al. Network representation learning: A survey[J]. IEEE transactions on Big Data, 2018.
- [20] Goyal P, Ferrara E. Graph embedding techniques, applications, and performance: A survey[J]. Knowledge-Based Systems, 2018, 151: 78-94.
- [21] 涂存超, 杨成, 刘知远, et al. 网络表示学习综述[J]. 中国科学: 信息科学, 2017, 8.
- [22] Battaglia P W, Hamrick J B, Bapst V, et al. Relational inductive biases, deep learning, and graph networks[J]. arXiv preprint arXiv:1806.01261, 2018.
- [23] Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint arXiv:1301.3781, 2013.
- [24] Perozzi B, Al-Rfou R, Skiena S. Deepwalk: Online learning of social representations[C]. Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, 2014: 701-710.
- [25] Grover A, Leskovec J. node2vec: Scalable feature learning for networks[C]. Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining, 2016: 855-864.
- [26] Wang D, Cui P, Zhu W. Structural deep network embedding[C]. Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining, 2016: 1225-1234.
- [27] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[J]. arXiv preprint arXiv:1609.02907, 2016.
- [28] Veličković P, Cucurull G, Casanova A, et al. Graph attention networks[J]. arXiv preprint arXiv:1710.10903, 2017.
- [29] Wu Z, Pan S, Chen F, et al. A comprehensive survey on graph neural networks[J]. arXiv preprint arXiv:1901.00596, 2019.
- [30] Zhang Z, Cui P, Zhu W. Deep learning on graphs: A survey[J]. arXiv preprint arXiv:1812.04202, 2018.

- [31] Zhou J, Cui G, Zhang Z, et al. Graph neural networks: A review of methods and applications[J]. arXiv preprint arXiv:1812.08434, 2018.
- [32] Zhu D, Cui P, Zhang Z, et al. High-order proximity preserved embedding for dynamic networks[J]. IEEE Transactions on Knowledge and Data Engineering, 2018, 30(11): 2134-2144.
- [33] De Lathauwer L, De Moor B, Vandewalle J. A multilinear singular value decomposition[J]. SIAM journal on Matrix Analysis and Applications, 2000, 21(4): 1253-1278.
- [34] Zuo Y, Liu G, Lin H, et al. Embedding temporal network via neighborhood formation[C]. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018: 2857-2866.
- [35] Mei H, Eisner J M. The neural hawkes process: A neurally self-modulating multivariate point process[C]. Advances in Neural Information Processing Systems, 2017: 6754-6764.
- [36] Sun Y, Han J, Yan X, et al. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks[J]. Proceedings of the VLDB Endowment, 2011, 4(11): 992-1003.
- [37] Dong Y, Chawla N V, Swami A. metapath2vec: Scalable representation learning for heterogeneous networks[C]. Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, 2017: 135-144.
- [38] Shi C, Hu B, Zhao W X, et al. Heterogeneous information network embedding for recommendation[J]. IEEE Transactions on Knowledge and Data Engineering, 2018, 31(2): 357-370.
- [39] Zhu R, Zhao K, Yang H, et al. AliGraph: a comprehensive graph neural network platform[J]. Proceedings of the VLDB Endowment, 2019, 12(12): 2094-2105.
- [40] Liu Y, Ge K, Zhang X, et al. Real-time Attention Based Look-alike Model for Recommender System[C]. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019: 2765-2773.
- [41] Ng A Y, Jordan M I, Weiss Y. On spectral clustering: Analysis and an algorithm[C]. Advances in neural information processing systems, 2002: 849-856.
- [42] Roweis S T, Saul L K. Nonlinear dimensionality reduction by locally linear embedding[J]. science, 2000, 290(5500): 2323-2326.
- [43] Belkin M, Niyogi P. Laplacian eigenmaps and spectral techniques for embedding and clustering[C]. Advances in neural information processing systems, 2002: 585-591.

- [44] Cao S, Lu W, Xu Q. Grarep: Learning graph representations with global structural information[C]. Proceedings of the 24th ACM international on conference on information and knowledge management, 2015: 891-900.
- [45] Ou M, Cui P, Pei J, et al. Asymmetric transitivity preserving graph embedding[C]. Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining, 2016: 1105-1114.
- [46] Tang J, Qu M, Wang M, et al. Line: Large-scale information network embedding[C]. Proceedings of the 24th international conference on world wide web, 2015: 1067-1077.
- [47] Yang C, Liu Z, Zhao D, et al. Network representation learning with rich text information[C]. Twenty-Fourth International Joint Conference on Artificial Intelligence, 2015.
- [48] Tu C, Liu H, Liu Z, et al. Cane: Context-aware network embedding for relation modeling[C]. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2017: 1722-1731.
- [49] Xu L, Wei X, Cao J, et al. Embedding of Embedding (EOE) Joint Embedding for Coupled Heterogeneous Networks[C]. Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, 2017: 741-749.
- [50] Tang J, Qu M, Mei Q. Pte: Predictive text embedding through large-scale heterogeneous text networks[C]. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015: 1165-1174.
- [51] Huang Z, Mamoulis N. Heterogeneous information network embedding for meta path based proximity[J]. arXiv preprint arXiv:1701.05291, 2017.
- [52] Fu T-Y, Lee W-C, Lei Z. Hin2vec: Explore meta-paths in heterogeneous information networks for representation learning[C]. Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 2017: 1797-1806.
- [53] Shang J, Qu M, Liu J, et al. Meta-path guided embedding for similarity search in large-scale heterogeneous information networks[J]. arXiv preprint arXiv:1610.09769, 2016.
- [54] Wang H, Zhang F, Hou M, et al. Shine: Signed heterogeneous information network embedding for sentiment link prediction[C]. Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, 2018: 592-600.

- [55] Zhang J, Xia C, Zhang C, et al. BL-MNE: emerging heterogeneous social network embedding through broad learning with aligned autoencoder[C]. 2017 IEEE International Conference on Data Mining (ICDM), 2017: 605-614.
- [56] Chang S, Han W, Tang J, et al. Heterogeneous network embedding via deep architectures[C]. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015: 119-128.
- [57] Wang X, Ji H, Shi C, et al. Heterogeneous graph attention network[C]. The World Wide Web Conference, 2019: 2022-2032.
- [58] Leskovec J, Kleinberg J, Faloutsos C. Graphs over time: densification laws, shrinking diameters and possible explanations[C]. Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, 2005: 177-187.
- [59] Aggarwal C, Subbian K. Evolutionary network analysis: A survey[J]. ACM Computing Surveys (CSUR), 2014, 47(1): 1-36.
- [60] Snijders T A. The statistical evaluation of social network dynamics[J]. Sociological methodology, 2001, 31(1): 361-395.
- [61] Mahajan V, Muller E, Bass F M. New product diffusion models in marketing: A review and directions for research[J]. Journal of marketing, 1990, 54(1): 1-26.
- [62] Anderson R M, Anderson B, May R M. Infectious diseases of humans: dynamics and control[M]. Oxford university press, 1992.
- [63] Zang C, Cui P, Faloutsos C. Beyond sigmoids: The net tide model for social network growth, and its applications[C]. Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016: 2015-2024.
- [64] Zhang T, Cui P, Faloutsos C, et al. Come-and-go patterns of group evolution: A dynamic model[C]. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016: 1355-1364.
- [65] Li J, Dani H, Hu X, et al. Attributed network embedding for learning in a dynamic environment[C]. Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 2017: 387-396.
- [66] Du L, Wang Y, Song G, et al. Dynamic Network Embedding: An Extended Approach for Skip-gram based Network Embedding[C]. IJCAI, 2018: 2086-2092.
- [67] Zhou L, Yang Y, Ren X, et al. Dynamic network embedding by modeling triadic closure process[C]. Thirty-Second AAAI Conference on Artificial Intelligence, 2018.

- [68] Trivedi R, Farajtabar M, Biswal P, et al. Representation learning over dynamic graphs[J]. arXiv preprint arXiv:1803.04051, 2018.
- [69] Zhao H, Yao Q, Li J, et al. Meta-graph based recommendation fusion over heterogeneous information networks[C]. Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017: 635-644.
- [70] Zhang Y, Ai Q, Chen X, et al. Joint representation learning for top-n recommendation with heterogeneous information sources[C]. Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 2017: 1449-1458.
- [71] Han X, Shi C, Wang S, et al. Aspect-Level Deep Collaborative Filtering via Heterogeneous Information Networks[C]. IJCAI, 2018: 3393-3399.
- [72] Hu B, Shi C, Zhao W X, et al. Leveraging meta-path based context for top-n recommendation with a neural co-attention model[C]. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018: 1531-1540.
- [73] Li A, Cornelius S P, Liu Y-Y, et al. The fundamental advantages of temporal networks[J]. Science, 2017, 358(6366): 1042-1046.
- [74] Chen H, Yin H, Wang W, et al. PME: projected metric embedding on heterogeneous networks for link prediction[C]. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018: 1177-1186.
- [75] Sun Y, Han J. Mining heterogeneous information networks: a structural analysis approach[J]. Acm Sigkdd Explorations Newsletter, 2013, 14(2): 20-28.
- [76] Yang J, Leskovec J. Community-affiliation graph model for overlapping network community detection[C]. 2012 IEEE 12th international conference on data mining, 2012: 1170-1175.
- [77] Wasserman S, Faust K. Social network analysis: Methods and applications[M]. 8. Cambridge university press, 1994.
- [78] Faust K. Centrality in affiliation networks[J]. Social networks, 1997, 19(2): 157-191.
- [79] Danielsson P-E. Euclidean distance mapping[J]. Computer Graphics and image processing, 1980, 14(3): 227-248.
- [80] Hsieh C-K, Yang L, Cui Y, et al. Collaborative metric learning[C]. Proceedings of the 26th international conference on world wide web, 2017: 193-201.

- [81] Bordes A, Usunier N, Garcia-Duran A, et al. Translating embeddings for modeling multi-relational data[C]. Advances in neural information processing systems, 2013: 2787-2795.
- [82] Lin Y, Liu Z, Sun M, et al. Learning entity and relation embeddings for knowledge graph completion[C]. Twenty-ninth AAAI conference on artificial intelligence, 2015.
- [83] Hussein R, Yang D, Cudré-Mauroux P. Are Meta-Paths Necessary? Revisiting Heterogeneous Graph Embeddings[C]. Proceedings of the 27th ACM International Conference on Information and Knowledge Management, 2018: 437-446.
- [84] Maaten L V D, Hinton G. Visualizing data using t-SNE[J]. Journal of machine learning research, 2008, 9(Nov): 2579-2605.
- [85] Golub G H, Van Loan C F. Matrix computations[M]. 3. JHU press, 2012.
- [86] Trefethen L N, Bau Iii D. Numerical linear algebra[M]. 50. Siam, 1997.
- [87] Kato T. Perturbation theory for linear operators[M]. 132. Springer Science & Business Media, 2013.
- [88] Parlett B N. The symmetric eigenvalue problem[M]. 20. siam, 1998.
- [89] Ritzer G, Stepnisky J. Modern sociological theory[M]. Sage publications, 2017.
- [90] Jolly A. Lemur social behavior and primate intelligence[J]. Science, 1966, 153(3735): 501-506.
- [91] Beale H D, Demuth H B, Hagan M. Neural network design[J]. Pws, Boston, 1996.
- [92] Yang B, Lei Y, Liu J, et al. Social collaborative filtering by trust[J]. IEEE transactions on pattern analysis and machine intelligence, 2016, 39(8): 1633-1647.
- [93] Wu L, Sun P, Fu Y, et al. A neural influence diffusion model for social recommendation[C]. Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval, 2019: 235-244.

## 致谢

行文至此，意味着硕士研究生的三年学习生活即将画上句号。在这三年里，我得到了很多老师的悉心指导，舍友同学的帮助，以及家人的支持。在此，我想向他们表达感激之情，感谢他们的鼓励和支持。

首先，我要感谢我的研究生导师，北京邮电大学计算机学院教授石川老师。研究生三年里，石老师不仅在学术研究上指导我逐步前行，而且在生活处事上提供了一些建议。在学术研究上，石老师为我的研究工作提供了细致的指导和宝贵的建议。在开展第一个研究工作时，石老师耐心引导我适应学术研究过程，教会我如何开展学术研究。石老师不仅及时帮助我解决模型实验中遇到的问题，还认真细心地指导我学术论文的写作，让我受益颇多。在生活处事上，石老师也教会了我很多道理。在我论文屡投屡拒的时候，石老师耐心鼓励我不气馁不放弃，劝导我从多角度看待问题，目光长远，坚持努力。因此，非常感谢石川老师，感谢老师给予的指导、帮助和鼓励。

其次，我要感谢从研究生一年级至三年级的合作老师，包括清华大学计算机系副教授刘知远老师，北京邮电大学计算机学院助理教授胡琳梅老师、王啸老师，伊利诺伊大学芝加哥分校计算机科学系教授 Philip S. Yu 老师，以及新加坡管理大学信息系统学院助理教授方元老师。感谢刘老师和胡老师为第一个研究工作提供的非常多的建议和意见；感谢王啸老师和 Philip S. Yu 老师为第二、三个研究工作提供的宝贵讨论和交流；感谢方老师为第四、五个研究工作提供的具体有效的建议、同时为我在新加坡交流提供的生活保障和关照。还要感谢腾讯微信事业群的实习导师们，包括谢若冰研究员、王伟研究员和张旭研究员等，感谢他们为我的第四个研究工作提供的数据平台支持以及模型构建方面的讨论。

然后，我要感谢同门师兄弟姐妹及舍友同学。在研究工作中，同门师兄弟姐妹为我提供了许多有建设性的讨论和交流，帮助我快速成长和学习，感谢他们的建议和帮助。同时，舍友同学也为我的研究工作提供了许多建设性意见和建议，并为我创造了舒适的环境，提供了学习之外的乐趣。感谢舍友同学的关照和包容。

最后，我要感谢家人的支持和鼓励。求学路上，父母家人始终如一的强有力支持和鼓励，是我前行过程中的重要支撑和动力。在物质上，他们提供有力保障，使我能够安心学习和科研；精神上，他们始终支持尊重我的选择，使得我能够专注于自己想做的事情。硕士研究生期间，一如既往的支持保证了我的学习和生活，为我的研究工作提供了重要的后勤保障。同时，感谢努力拼搏的自己。

## 攻读学位期间取得的研究成果

### 会议论文

- [1] **Lu Yuanfu**, Fang Y, Shi C. Meta-learning on Heterogeneous Information Networks for Cold-start Recommendation[C]. Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020. (CCF-A类, EI 检索)
- [2] **Lu Yuanfu**, Shi C, Hu L, Liu Z. Relation Structure-aware Heterogeneous Information Network Embedding[C]. Proceedings of the AAAI Conference on Artificial Intelligence. 2019, 33: 4456-4463. (CCF-A类, EI 检索)
- [3] **Lu Yuanfu**, Wang X, Shi C, et al. Temporal Network Embedding with Micro-and Macro-dynamics[C]. Proceedings of the 28th ACM International Conference on Information and Knowledge Management. 2019: 469-478. (CCF-B类, EI 检索)
- [4] **Lu Yuanfu**, Xie R, Shi C, et al. Social Influence Attentive Neural Network for Friend-Enhanced Recommendation[C]. Proceedings of ECML-PKDD. 2020. (CCF-B类, EI 检索)

### 期刊论文

- [5] Shi C, **Lu Yuanfu**, Hu L, Liu Z, Ma H. RHINE: Relation Structure-Aware Heterogeneous Information Network Embedding[J]. IEEE Transactions on Knowledge and Data Engineering, 2020. (CCF-A类, SCI 检索)
- [6] Wang X, **Lu Yuanfu**, Shi C. et al. Dynamic Heterogeneous Information Network Embedding with Meta-path based Proximity[J]. IEEE Transactions on Knowledge and Data Engineering, 2020. (CCF-A类, SCI 检索)

### 专利

- [7] 石川、陆元福、胡琳梅. 一种异质信息网络嵌入方法、装置、电子设备及存储介质. 201910052260.6.
- [8] 王啸、石川、陆元福. 一种动态异质信息网络嵌入方法、装置和电子设备. 201910280097.9.
- [9] 王伟, 陆元福, 谢若冰. 一种微信朋友在看场景下评估用户互动及多重影响力的技术方案. 2019110959154.