**中图分类号：TP391**

**论文编号：10006SY1906428**



硕 士 学 位 论 文

**基于图神经网络的异构图**

**表示学习和推荐算法研究**

作者姓名 赵正阳

学科专业 软件工程

指导教师 王德庆 副研究员

培养学院 计算机学院

**Research on Heterogeneous Graph Representation Learning and Recommendation Algorithm Based on Graph Neural Network**

A Dissertation Submitted for the Degree of Master

**Candidate: Zhao Zhengyang**

**Supervisor: Associate Prof. Wang Deqing**

School of Computer Science and Engineering

Beihang University, Beijing, China

**中图分类号：TP391**

**论文编号：10006SY1906428**

硕 士 学 位 论 文

**基于图神经网络的异构图**

**表示学习和推荐算法研究**

作者姓名 赵正阳 申请学位级别 全日制学术型硕士

指导教师姓名 王德庆 职 称 副研究员

学科专业 软件工程 研究方向 数据挖掘

学习时间自 2019 年 9 月 1 日 起至 2021 年 12 月 11 日止

论文提交日期 2021 年 12 月 14 日 论文答辩日期 2021 年 12 月 11 日

学位授予单位 北京航空航天大学 学位授予日期 2022 年 1 月 11 日

关于学位论文的独创性声明

本人郑重声明：所呈交的论文是本人在指导教师指导下独立进行研究工作所取得的成果，论文中有关资料和数据是实事求是的。尽我所知，除文中已经加以标注和致谢外，本论文不包含其他人已经发表或撰写的研究成果，也不包含本人或他人为获得北京航空航天大学或其它教育机构的学位或学历证书而使用过的材料。与我一同工作的同志对研究所做的任何贡献均已在论文中作出了明确的说明。

若有不实之处，本人愿意承担相关法律责任。

学位论文作者签名：        日期： 年 月 日

学位论文使用授权书

本人完全同意北京航空航天大学有权使用本学位论文（包括但不限于其印刷版和电子版），使用方式包括但不限于：保留学位论文，按规定向国家有关部门（机构）送交学位论文，以学术交流为目的赠送和交换学位论文，允许学位论文被查阅、借阅和复印，将学位论文的全部或部分内容编入有关数据库进行检索，采用影印、缩印或其他复制手段保存学位论文。

保密学位论文在解密后的使用授权同上。

学位论文作者签名： 日期： 年 月 日

指导教师签名： 日期： 年 月 日

**摘 要**

随着互联网技术的快速发展，互联网上的科研文献数量呈现爆炸式增长。快速增长的文献数据给研究者带来便利的同时也使得从海量文献中挖掘有价值的信息变得极其困难。学术网络由学者、论文、期刊等不同类型的顶点组成，是一种典型的异构图，而异构图表示学习及其在学术领域的应用是目前的研究热点。图神经网络作为一种基于深度学习的图表示学习技术吸引了大量的研究兴趣，但是仍然面临着可扩展性差、输入特征简单等挑战。同时，现有的学术推荐算法大多基于关键词匹配和引用数统计分析实现，存在应用范围有限和冷启动问题。

针对上述问题，本文对基于图神经网络的异构图表示学习及其在推荐算法中的应用进行了研究，主要工作如下：

（1）针对异构图表示学习任务，提出了基于对比学习的关系感知异构图神经网络模型RHCO。该模型在顶点嵌入中融入了关系信息，并利用预训练的异构图Transformer (HGT)计算的注意力权重选择正样本，使得模型具有可扩展性，最后通过标签平滑处理提升模型在顶点分类任务上的性能。实验结果表明，RHCO模型在顶点分类任务准确率上相比最新的GNN模型取得了2.1%的提升。

（2）针对领域学者排名的学术推荐任务，提出了基于图神经网络的学术推荐算法GARec。该算法通过论文标题和关键词的对比学习对预训练的SciBERT模型进行微调，从而获得论文标题向量，利用论文标题向量计算查询词与论文的相似度得分，实现论文召回，最后利用RHCO模型学习到的学者顶点嵌入计算查询词与学者的相似度得分，实现学者排名。定量实验和案例分析证明了该算法的有效性。

（3）设计并实现了学术推荐系统，使用微软学术数据构造了一个大规模学术网络数据集，包含学者、论文、期刊、领域和机构顶点，集成上述两个研究实现了领域学者排名功能的可视化呈现。

**关键词：**图神经网络，异构图，图表示学习，推荐算法

**Abstract**

With the rapid development of Internet technology, the number of scientific research literature on the Internet has grown explosively. The rapid growth of literature data brings convenience to researchers, but also makes it extremely difficult to mine valuable information from massive literature. Bibliographic network is a typical heterogeneous graph, consisting of different types of nodes such as authors, papers, venues, etc. Heterogeneous graph represents learning and its application in the academic field is a current research hotspot. Graph neural network, as a graph representation learning technique based on deep learning, has attracted a lot of research interest. But it still faces challenges such as poor scalability and simple input features. Most existing academic recommendation algorithms are based on keyword matching and statistical analysis of citation, which leads to limited application scope and cold start problems.

To solve the above problems, this paper studies heterogeneous graph representation learning based on graph neural network and its application in recommendation algorithms. The main work is as follows:

1) For heterogeneous graph representation learning task, this paper proposes Relation-aware Heterogeneous Graph Neural Network with Contrastive Learning (RHCO). This model incorporates relational information into node embeddings, then utilizes attention weight calculated by the pretrained Heterogeneous Graph Transformer (HGT) to select positive samples, which makes the model scalable, and improves performance on node classification through label smoothing at last. Experimental results show that RHCO achieves 2.1% improvement over the state-of-the-art GNN model on node classification.

2) For academic recommendation task of scholar ranking by field, this paper proposes an academic recommendation algorithm based on graph neural network (GARec). The algorithm fine-tunes pre-trained SciBERT model through contrastive learning of paper titles and keywords, so as to obtain vectors of paper titles, then uses the paper title vectors to calculate similarity score between query word and papers, thus realizing the paper recall, and utilizes scholar node embeddings learned by the RHCO model to calculate similarity score between query word and scholars, thus realizing scholar ranking. Quantitative experiments and case study prove the effectiveness of the algorithm.

3) This paper designs and implements an academic recommender system; constructs a large-scale bibliographic network from Microsoft Academic Graph data, consisting of authors, papers, venues, fields and institutions, and integrates the above two studies to realize visualization of scholar ranking by field.

**Keywords:** Graph Neural Network, Heterogeneous Graph, Graph Representation Learning, Recommendation Algorithm

**目 录**

[第一章 绪论 1](#_Toc89262097)

[1.1 研究背景与意义 1](#_Toc89262098)

[1.2 国内外研究现状 2](#_Toc89262099)

[1.2.1 异构图表示学习 2](#_Toc89262100)

[1.2.2 基于图神经网络的推荐算法 5](#_Toc89262101)

[1.3 研究目标与内容 8](#_Toc89262102)

[1.4 论文结构与内容安排 8](#_Toc89262103)

[第二章 理论基础 10](#_Toc89262104)

[2.1 图神经网络 10](#_Toc89262105)

[2.1.1 消息传递 10](#_Toc89262106)

[2.1.2 异构图 11](#_Toc89262107)

[2.1.3 小批量训练和邻居采样 13](#_Toc89262108)

[2.1.4 下游任务 14](#_Toc89262109)

[2.2 自监督学习和对比学习 15](#_Toc89262110)

[2.2.1 自监督学习 15](#_Toc89262111)

[2.2.2 对比学习 16](#_Toc89262112)

[2.3 词向量模型 16](#_Toc89262113)

[2.3.1 Skip-gram模型 16](#_Toc89262114)

[2.3.2 在图表示学习领域的应用 18](#_Toc89262115)

[2.4 小结 19](#_Toc89262116)

[第三章 基于对比学习的异构图神经网络模型 20](#_Toc89262117)

[3.1 引言 20](#_Toc89262118)

[3.2 基于对比学习的关系感知异构图神经网络 21](#_Toc89262119)

[3.2.1 输入特征转换 22](#_Toc89262120)

[3.2.2 网络结构编码器 22](#_Toc89262121)

[3.2.3 正样本选择策略 23](#_Toc89262122)

[3.2.4 正样本图编码器 24](#_Toc89262123)

[3.2.5 损失函数 25](#_Toc89262124)

[3.2.6 模型分析 26](#_Toc89262125)

[3.3 实验设计与分析 26](#_Toc89262126)

[3.3.1 数据集 26](#_Toc89262127)

[3.3.2 对比模型 28](#_Toc89262128)

[3.3.3 实现细节 28](#_Toc89262129)

[3.3.4 评价指标 28](#_Toc89262130)

[3.3.5 实验结果 29](#_Toc89262131)

[3.3.6 参数敏感性分析 30](#_Toc89262132)

[3.3.7 消融实验 31](#_Toc89262133)

[3.4 小结 32](#_Toc89262134)

[第四章 基于图神经网络的学术推荐算法 33](#_Toc89262135)

[4.1 问题定义 33](#_Toc89262136)

[4.2 算法设计 33](#_Toc89262137)

[4.2.1 预训练论文向量 34](#_Toc89262138)

[4.2.2 构造学者排名 35](#_Toc89262139)

[4.2.3 论文召回 36](#_Toc89262140)

[4.2.4 精排 36](#_Toc89262141)

[4.2.5 推断 38](#_Toc89262142)

[4.3 实验设计与分析 38](#_Toc89262143)

[4.3.1 数据集 38](#_Toc89262144)

[4.3.2 对比模型 40](#_Toc89262145)

[4.3.3 实现细节 40](#_Toc89262146)

[4.3.4 评价指标 40](#_Toc89262147)

[4.3.5 搜索论文 41](#_Toc89262148)

[4.3.6 学者排名 43](#_Toc89262149)

[4.3.7 参数敏感性分析 44](#_Toc89262150)

[4.3.8 案例分析 45](#_Toc89262151)

[4.4 小结 46](#_Toc89262152)

[第五章 学术推荐系统设计与实现 48](#_Toc89262153)

[5.1 系统概述 48](#_Toc89262154)

[5.2 总体设计 49](#_Toc89262155)

[5.3 详细设计 51](#_Toc89262156)

[5.3.1 数据库设计 51](#_Toc89262157)

[5.3.2 用例图 51](#_Toc89262158)

[5.3.3 论文召回模块 52](#_Toc89262159)

[5.3.4 学者排名模块 52](#_Toc89262160)

[5.4 效果展示 52](#_Toc89262161)

[5.5 小结 56](#_Toc89262162)

[总结与展望 57](#_Toc89262163)

[总结 57](#_Toc89262164)

[展望 57](#_Toc89262165)

[参考文献 59](#_Toc89262166)

[攻读硕士学位期间取得的学术成果 64](#_Toc89262167)

[致谢 65](#_Toc89262168)

**图 目**

[图 1 异构学术网络结构 1](#_Toc89262169)

[图 2 领域学者排名功能 2](#_Toc89262170)

[图 3 消息传递过程 10](#_Toc89262171)

[图 4 异构学术网络示例 12](#_Toc89262172)

[图 5 两层消息传递邻居采样示例 13](#_Toc89262173)

[图 6 CBOW和Skip-gram模型结构 17](#_Toc89262174)

[图 7 词向量包含的语义规律 18](#_Toc89262175)

[图 8 DeepWalk算法 19](#_Toc89262176)

[图 9 RHCO模型结构 21](#_Toc89262177)

[图 10 正样本选择策略 24](#_Toc89262178)

[图 11 RHCO模型参数敏感性分析实验结果 30](#_Toc89262179)

[图 12 RHCO模型消融实验结果 31](#_Toc89262180)

[图 13 GARec算法整体框架 34](#_Toc89262181)

[图 14 通过对比学习微调SciBERT模型 35](#_Toc89262182)

[图 15 oag-cs数据集网络结构 39](#_Toc89262183)

[图 16 学者排名实验结果 43](#_Toc89262184)

[图 17 GARec算法参数敏感性分析实验结果 44](#_Toc89262185)

[图 18 MTV模式操作流程图 48](#_Toc89262186)

[图 19 学术推荐系统架构图 49](#_Toc89262187)

[图 20 系统E-R图 50](#_Toc89262188)

[图 21 系统用例图 51](#_Toc89262189)

[图 22 系统流程图 52](#_Toc89262190)

[图 23 用户注册页面 53](#_Toc89262191)

[图 24 用户登录页面 53](#_Toc89262192)

[图 25 系统首页 53](#_Toc89262193)

[图 26 论文搜索结果 54](#_Toc89262194)

[图 27 论文详情页面 55](#_Toc89262195)

[图 28 学者搜索结果 55](#_Toc89262196)

[图 29 学者详情页面 56](#_Toc89262197)

**表 目**

[表 1 异构图表示学习数据集 27](#_Toc89262198)

[表 2 顶点分类实验结果 29](#_Toc89262199)

[表 3 oag-cs数据集 39](#_Toc89262200)

[表 4 学者排名示例数据 39](#_Toc89262201)

[表 5 论文搜索结果示例 42](#_Toc89262202)

[表 6 学者排名实验结果 43](#_Toc89262203)

[表 7 “计算机视觉”学者排名案例分析 45](#_Toc89262204)

[表 8 “基于图神经网络的推荐系统”学者排名案例分析 46](#_Toc89262205)

# 绪论

## 研究背景与意义

互联网技术的快速发展推动了大数据时代的到来，使得人们获取信息更加便捷。同时，爆炸式增长的信息也带来了一些挑战。IDC发布的《数据时代2025》报告显示，全球的数据量将从2019年的45 ZB增长到2025年的175 ZB。如何对数量如此庞大的数据进行分析，并从中挖掘对用户有价值的信息变得越来越重要。在现实生活中，图结构的数据无处不在，例如社交网络、学术网络等。在这些场景中，使用图能够直观、清晰地刻画实体之间的关系。虽然图结构的数据对人类来说比较直观，但对计算机来说是难以理解的。因此，图表示学习成为近年来的一个研究热点。图表示学习旨在将图中的顶点转换为低维的嵌入向量，同时保留图的拓扑结构信息。这些嵌入向量可用于多种下游任务，例如顶点分类、连接预测、推荐等。图神经网络(graph neural network, GNN)[1]作为一种基于深度学习的强大的图表示学习技术吸引了大量的研究兴趣。然而，真实世界的网络通常具有多种类型的实体和关系，因此异构图(heterogeneous graph)被提出并广泛用于多种应用场景[2]。例如，图 1所示的异构学术网络包含学者、论文、期刊、领域四种顶点以及这些顶点之间的四种关系。异构图包含更加丰富的语义信息，同时其异构性也使得表示学习更加具有挑战性。



图 1 异构学术网络结构

随着科学技术不断进步，在过去的几个世纪中，科研文献的数量呈指数级增长，每12年增加一倍，并且每个月都有几百万篇新的文献发表[3]。如何从海量的科研文献数据中检索出用户感兴趣的内容，从而实现学术推荐功能变得越来越重要。例如，Google学术[[1]](#footnote-2)、微软学术[[2]](#footnote-3)和科搜[[3]](#footnote-4)等学术搜索引擎都提供了领域学者排名功能，即输入一个领域关键词，检索出该领域最相关的学者并对学者进行排名，如图 2所示。但是，现有的学者排名功能大多是基于关键词匹配和引用数统计分析实现的，但这种方法无法充分利用学术网络所包含的结构和语义信息对学者、论文等实体进行表征，推荐准确度较差。因此，异构图的表示学习在学术推荐中的应用是一个值得研究的课题。



图 2 领域学者排名功能

## 国内外研究现状

### 异构图表示学习

异构图表示学习方法主要分为基于随机游走的方法[5-9]和基于图神经网络的方法[10-22]两大类。前者通过随机游走策略生成顶点序列，使用语言建模模型Skip-gram学习顶点嵌入；后者基于消息传递思想[4]，通过聚集邻居特征更新顶点特征的方式学习顶点嵌入。

基于随机游走的图表示学习方法起源于Perozzi等人在2014年提出的DeepWalk[5]，首先在图中进行给定数量和长度的随机游走，之后将顶点视为单词、顶点序列视为句子、随机游走结果视为语料库，使用语言建模模型Skip-gram[23, 24]以最大化单词上下文（顶点邻居）的似然概率为目标学习词向量（顶点嵌入）。Grover等人在2016年提出了node2vec[6]，在DeepWalk的基础上进行了改进，提出了有偏的二阶随机游走策略。这种策略的优点是比较灵活，能够通过两个超参数控制随机游走的偏向性，从而适应不同的图。Yuxiao Dong等人在2017年提出了异构图表示学习方法metapath2vec[8]，该方法使用基于元路径的随机游走策略来采样异构图的顶点序列。基于元路径的随机游走能够保证不同类型顶点之间的语义关系被结合到Skip-gram中。Taoyang Fu等人提出的异构图表示学习方法HIV2Vec[9]考虑两个顶点之间存在元路径实例的概率，使用随机游走来产生训练样本，使用一个单隐藏层的前馈神经网络来训练二分类器。

图神经网络(graph neural network, GNN)是图表示学习的另一个重要的研究分支。GNN的概念最早由Scarselli等人[1]在2008年提出，将神经网络扩展到图领域。GNN的核心思想是图中每个顶点的状态由其本身的特征和邻居的状态聚集得到，可以视为消息传递框架[4]的特例。近年来，研究者提出了以消息传递为理论基础的各种GNN变体。

Kipf等人在2017年提出了图卷积网络(Graph Convolutional Network, GCN)[10]。图卷积操作是通过将传统傅立叶变换类比到图领域，对图的拉普拉斯矩阵进行谱分解，并使用一阶切比雪夫多项式近似来定义的，最后一层输出的隐含状态即为顶点的嵌入向量。

Kipf等人在2018年提出了Relational GCN (R-GCN)[11]模型，将GCN扩展到异构图上，在消息传递过程中考虑了边的异构性，同种类型的邻居分别聚集并归一化。

Veličković等人在2018年提出的图注意力网络(Graph Attention Network, GAT)[13]将注意力机制应用到图神经网络，通过对每个顶点应用注意力机制来学习不同邻居顶点的重要性，并将其作为权重计算邻居特征的线性组合来更新顶点特征。

Zhang Chuxu等人在2019年提出了HetGNN[14]，该模型假设每个顶点关联了不同类型的内容属性（ID、文本、图像等）。首先将每个顶点的内容属性编码为一个向量，之后将每个顶点不同类型的邻居的内容嵌入分别聚集为一个向量，其中邻居集合通过有重启的随机游走策略采样得到。最后，使用与GAT类似的注意力机制将不同类型的邻居嵌入组合起来，得到最终的顶点嵌入。

Wang Xiao等人在2019年提出的异构图注意力网络(Heterogeneous Graph Attention Network, HAN)[15]在GAT的基础上进行了改进。HAN利用元路径来建模顶点之间的高阶邻接关系，由顶点层次的注意力和语义层次的注意力两部分组成。对于顶点层次的注意力，给定一条元路径*P*，使用与GAT相同的方式学习每个顶点基于元路径*P*的嵌入；语义层次的注意力将基于不同元路径的顶点嵌入组合为最终顶点嵌入。

Xinyu Fu 等人在2020年提出的MAGNN[16]模型在HAN的基础上进行了改进，由元路径内的聚集和元路径间的聚集两部分组成。元路径内的聚集将HAN顶点层次的注意力模块中对基于元路径的邻居特征的聚集改为对元路径实例特征的聚集，从而利用了元路径中间顶点的信息，元路径实例的特征是通过元路径实例编码器将所有中间顶点的特征转换为一个向量得到。元路径间的聚集使用注意力机制将每个顶点关于不同元路径的嵌入组合起来，方法与HAN语义层次的注意力相同。

Hu Ziniu等人在2020年提出了异构图Transformer模型(HGT)[17]，使用每条边的类型来参数化类似Transformer[25, 26]的自注意力，从而可以使用相等甚至更少的参数捕捉到不同关系的公共模式和特定模式。此外，与需要手动设计元路径的方法不同，HGT可以通过多层GNN自动学习隐式元路径的重要性，以合并高阶异构邻居信息。

Yu Le等人在2020年提出的HGConv模型[18]使用混合微观、宏观层次卷积的结构。微观层次卷积用于学习同种关系（边类型）下顶点的重要性；宏观层次卷积用于学习不同关系的重要性；加权残差连接用于组合顶点继承自上一层的属性和通过微观、宏观层次卷积聚集得到的邻居信息。混合微观、宏观层次卷积使得HGConv能够充分利用顶点和关系的异构信息，并且具有可解释性。

Yu Le等人在2021年提出的R-HGNN模型[19]在HGConv的基础上进行了改进。特定关系的顶点表示学习将输入的异构图分解为关系二分图，使用图卷积分别从每个关系子图学习不同的顶点表示；增加了关系表示学习模块，逐层学习关系的表示，用于指导顶点表示学习过程；最后使用关系感知的表示混合模块将关系感知的顶点表示聚集为一个表示，考虑关系的语义特征。

Qian Huang等人在2020年提出的C&S模型[20]将经典的标签传播算法[21]与GNN结合。首先使用不考虑图结构的简单线性模型进行基础预测；之后进行两种类型的标签传播：一个通过建模相关误差来修正基础预测，另一个平滑最终预测。C&S的标签传播只是后处理步骤，训练流程不是端到端的。另外，图只用在这两个后处理步骤中，并不用于基础预测，这使得与标准GNN模型相比参数更少、训练更快、更具可扩展性。

Wang Xiao等人在2021年将对比学习应用到GNN领域，提出了使用跨视图对比学习的自监督异构图神经网络HeCo[22]。该模型分别使用网络结构编码器和元路径编码器捕获图的局部结构和高阶结构，通过这两个视图学习到的顶点嵌入进行对比学习。该模型首次将对比学习方法应用到GNN领域，但过度依赖于元路径，因此不可扩展。

现有的方法主要有两个不足：

（1）**顶点的输入特征简单。**HGT, HGConv, R-HGNN等模型需要所有类型的顶点都有输入特征，而大部分方法使用的异构图数据集只有一种类型的顶点（通常是“论文”）有输入特征，通常是简单的关键词词袋表示或one-hot向量。为了获取其他类型顶点的输入特征，可采用邻居平均和预训练（例如使用metapath2vec）两种方法。

（2）**无法支持大规模数据集。**大部分方法使用的数据集都只有几千个顶点、几万条边，只有HGT和R-HGNN使用的数据集达到百万至千万规模。而有些模型采用的方法是不可扩展的（例如基于元路径的方法），当数据集过大时无法进行全图训练。

### 基于图神经网络的推荐算法

随着互联网上信息的爆炸式增长，推荐系统在缓解信息过载方面发挥着重要作用[28]。推荐系统从用户-物品交互和物品属性中推断用户喜好，并进一步推荐用户可能感兴趣的物品[29]。由于很多数据本质上具有图结构，而GNN在图表示学习方面具有优势，因此在推荐系统中利用图神经网络成为了一个热点研究领域。根据是否考虑物品顺序，推荐系统可以分为一般推荐[30-40]和顺序推荐[41-45]；根据使用的信息类型进一步分为无辅助信息（仅使用用户-物品交互）、使用社交关系信息和使用知识图谱三类[28]。

（1）一般推荐

一般推荐旨在利用用户-物品交互数据建模用户喜好，从而提供反映每个用户的长期喜好的静态推荐列表。然而，对于数据稀疏性和冷启动场景，推荐系统的性能会显著下降。为了解决这两个问题，研究者尝试利用辅助信息，例如社交网络和知识图谱。

推荐系统背后的关键数学问题是矩阵补全(matrix completion)。现有方法通过分解交互矩阵来学习用户和物品表示，并使用学习到的嵌入来推断用户对目标物品的喜好。从图的角度，交互数据可以被表示为用户-物品二分图，矩阵完成可以被认为是图上的连接预测。由于GNN在学习图数据方面的优越性，越来越多的研究利用GNN以图的形式捕获用户和物品之间的高阶交互关系

Berg等人在2017年提出的GC-MC[30]旨在解决评分预测问题，将交互数据表示为二分图，边标签表示已知评分。GC-MC仅使用与用户交互过的物品来建模用户顶点，忽略了原始用户表示，反之亦然。GC-MC采用平均池化来聚集邻居顶点，即假设不同物品对于用户喜好同等重要。GC-MC仅考虑了一阶邻居，没有充分利用通过图结构的消息传递。完全丢弃用户和物品的原始信息可能会忽略内在的用户喜好和物品属性。

Zhang, Jiani等人在2019年提出的STAR-GCN[31]堆叠了多个GCN块，每个块的结构都是相同的。堆叠GCN块而不是直接堆叠多层GCN的动机是堆叠过多的卷积层可能会带来过平滑问题。为了连接相邻的块，STAR-GCN引入了重建机制，旨在从聚集后的表示中恢复初始输入向量并作为下一个块的输入。STAR-GCN利用重建策略来缓解过平滑问题，联合优化预测任务和重建任务，使用最后一层的顶点表示作为评分预测。

Ying, Rex等人在2018年提出的PinSage[33]组合了随机游走策略和图卷积来学习顶点嵌入。PinSage设计了一种基于随机游走的采样方法来采样固定大小的邻居，而不是在全图上进行传播。归一化的访问次数被用作聚集邻居信息时的重要性，实验结果表明该策略比平均池化的性能更好。采样方法提升了效率，代价是丢失了原始图结构。邻居采样的随机性可能会影响模型性能。

随着在线社交网络的出现，社交推荐系统被提出，利用每个用户的局部邻居的喜好来缓解数据稀疏性，从而更好地建模用户嵌入。社交网络是一个同构图，每个用户是一个顶点，用户之间的边表示社交关系。实际上，社交影响在社交网络中递归地传播和扩散，即用户可能会受到他朋友的朋友的影响。GNN的一大特点是迭代传播，这与朋友之间的信息扩散过程是一致的。因此，研究人员开始利用GNN来模拟用户如何受到递归社交扩散过程的影响。

Wu, Le等人在2019年提出的DiffNet[34]从用户的社交关系和历史交互行为建模用户喜好。利用GraphSAGE框架来模拟用户如何受到递归社交扩散过程的影响。DiffNet假设朋友具有相同影响，使用平均池化来聚集两跳朋友的表示和一跳历史物品的表示，二者相加作为最终用户表示。局限性是相同影响假设可能与实际情况不符。另外，物品表示也可以使用用户表示来增强。

Fan, Wenqi等人在2019年提出的GraphRec[35]使用GAT分别学习社交网络和用户-物品二分图中的用户/物品表示，通过拼接用户在物品空间和社交空间的表示并应用MLP获取最终用户表示。注意力机制通过区分朋友的影响和物品的重要性提升了性能。但是由于GraphRec仅使用单层GAT在社交图中建模社交影响，社交影响可能没有得到充分利用。

Wu, Qitian等人在2019年提出的DANSER[36]考虑用户之间的社交关系和物品之间的关系。与大多数现有的假设来自朋友的社交影响是静态的方法不同，DANSER利用GAT学习两个社交影响的表示，一个由用户特定的注意力权重建模，另一个由动态的、上下文感知的注意力权重建模。类似地，每个物品也有两个属性表示。总体而言，该模型在集成了朋友影响下的用户静态和动态喜好，但它为物品之间的关系提供了太多自由。

知识图谱(knowledge graph, KG)由“头实体-关系-尾实体”三元组组成：，其中*E*和*R*分别是实体集合和关系集合。将知识图谱结合进推荐可以提高结果的精度、增加推荐物品的多样性以及为推荐系统带来可解释性。尽管知识图谱具有丰富的信息，但由于其复杂的图结构（即多种类型的实体和关系），在推荐系统中利用知识图谱相当具有挑战性。以前的工作通过知识图谱嵌入方法对知识图谱进行预处理以学习实体和关系的嵌入，或者设计元路径来聚集邻居信息。受GNN在图学习中的启发，研究者尝试利用GNN来捕获物品之间的相关性。

Wang, Hongwei等人在2019年提出的KGCN[38]利用用户特定的、关系感知的GNN来聚集邻居实体的信息。KGCN利用知识图谱获得语义感知的物品表示。为了计算效率，KGCN为每个实体采样固定数量的邻居作为其接收域(receptive field)。邻居的权重得分依赖于关系类型和特定用户，这既表征了KG的语义信息，又表征了用户在某个关系中的个性化兴趣。动机是不同用户会对不同关系有不同的重视程度。KGCN使用GAT作为传播算法，但给定特定用户和关系类型，邻居的重要性是静态的。物品的最终表示对于不同的用户是不同的，这从用户角度整合了知识图谱中的语义信息。

Wang, Hongwei等人在2019年提出的KGNN-LS[39]旨在通过识别对于给定用户重要的知识图谱关系来学习用户特定的物品嵌入。该模型的创新点是标签平滑正则化，假设知识图谱中相邻的物品可能具有相似的用户相关标签。KGNN-LS使用一个用户特定的关系打分函数来提供关系对用户的重要性。给定关系打分函数，知识图谱中的多个关系可以被表示为用户相关的邻接矩阵，将复杂的知识图谱转换为传统的图，从而可以使用GCN。此外，KGNN-LS还提出了一种标签平滑正则化，以帮助模型泛化到未观察到的交互。标签平滑等价于图上的标签传播。

Wang, Xiang等人在2019年提出的KGAT[40]将用户顶点视为知识图谱中的一种实体，将用户和物品之间的交互视为一种关系。即KGAT将两个图集成为一个图。这一方面统一了传播操作，另一方面也可能引入一些噪声。KGAT使用GAT机制充分挖掘实体之间的关系。邻居的重要性取决于两端的实体及其关系。另外，KGAT采用广泛使用的TransR方法，使用转移原理（头实体+关系≈尾实体）来规范实体和关系的嵌入。

（2）顺序推荐

顺序推荐基于用户最近的活动预测用户接下来的喜好，旨在建模连续物品中的顺序模式，并为用户生成适时的推荐。大多数现有方法仅从序列数据中推断时间喜好，即首先将序列数据转换为序列图，通过GNN方法捕获序列知识。

Wu, Shu等人在2019年提出的SR-GNN[41]为每个会话序列构造了一个有向会话图：连续被点击的两个物品之间存在一条有向边。SR-GNN定义了两个邻接矩阵，分别表示会话图中入边和出边的加权连接。在传播过程中，物品嵌分别被映射到入边表示和出边表示，即上一个和下一个被点击的物品传播到当前物品。SR-GNN假设物品的邻居具有相同的重要性，因为归一化的邻接矩阵为邻居赋予相同的权重。SR-GNN利用GRU机制更新当前物品嵌入。为了更好地预测用户的下一次点击，需要学习反映用户最近喜好的会话的嵌入。SR-GNN将GNN输出的最后一个点击物品的嵌入作为会话的局部嵌入，通过注意力机制聚集会话中所有物品的嵌入得到会话的全局嵌入，最后组合两个向量得到会话的最终嵌入。

Song, Weiping等人在2019年提出的DGRec[45]仅将社交网络视为图数据。使用LSTM机制从用户的最近会话行为中提取用户的动态兴趣。用户的表示在社交网络中是动态的。考虑到社交影响可能随着上下文变化，DGRec采用GAT来区分朋友的影响。消融实验表明，与使用社交网络的通用推荐和没有辅助信息的顺序推荐相比，DGRec的性能更出色。实验结果证明了在顺序推荐中考虑动态社交影响的合理性。

## 研究目标与内容

本文的研究目标是改进现有的基于图神经网络的异构图表示学习方法，提升模型在图挖掘任务上的性能；借助改进的异构图表示学习方法实现一种基于图神经网络的推荐算法，并将其应用到学术搜索引擎中，实现学术推荐功能。

本文的主要研究内容包括以下几个方面：

（1）研究基于图神经网络的异构图表示学习方法，在输入特征、模型结构、可扩展性等方面对现有模型进行改进。通过随机游走采样顶点序列并使用Skip-gram模型预训练的方式获得包含网络结构信息的输入特征；通过改进模型结构和训练方式使其能够适配小批量训练，从而能够应用于大规模数据集，并提升模型在顶点分类等基本图挖掘任务上的性能。

（2）研究基于图神经网络的推荐算法，使用上述改进的异构图表示学习模型获得“学者”、“论文”、“领域”等顶点的表示向量，设计一种基于图神经网络的学术推荐算法，使得算法能够综合考虑来自论文的语义信息以及来自网络的结构信息，提升推荐的准确性。

（3）构造一个较大规模的学术网络数据集，利用上述推荐算法实现根据输入的领域关键词对学者进行排名等功能，并实现可视化系统。

## 论文结构与内容安排

论文分为六章，内容安排如下：

第一章首先分析了论文的研究背景，以及异构图表示学习及其在推荐算法中应用的研究意义。接着详细介绍了异构图表示学习和基于图神经网络的推荐算法这两个领域的研究现状。最后对本文的研究目标与内容以及论文的内容安排做了说明。

第二章介绍了本文研究内容相关的理论基础知识。首先介绍了图神经网络的基本概念、消息传递模型以及常见的下游任务；接着介绍了自监督学习和对比学习的基本思想及其与图表示学习领域的结合；最后介绍了词向量模型及其在图表示学习领域的应用。

第三章提出了基于对比学习的异构图神经网络模型。首先分析了在异构图表示学习领域应用对比学习存在的问题以及现有方法的不足。接着介绍了本文提出的模型结构和主要的改进点。最后介绍了实验使用的数据集、评价指标和对比模型，并通过定量的实验数据展示了提出的模型在顶点分类任务上的性能提升。

第四章提出了基于图神经网络的学术推荐算法。首先介绍了本文研究的推荐问题定义以及与传统推荐任务的不同。接着介绍了本文提出的推荐算法流程。最后介绍了数据集的构造以及实验使用的评价指标，并通过定量的实验数据和案例分析展示了提出的算法的有效性。

第五章介绍了基于本文提出的学术推荐算法构建的学术推荐系统。首先介绍了系统概述和总体设计。接着介绍了各功能模块的详细设计以及论文召回和学者排名两个核心功能的实现方式。最后通过系统演示展示了推荐算法的效果。

最后是对本文工作的总结和对今后工作的展望。全面总结了本文提出的基于对比学习的异构图神经网络模型和基于图神经网络的推荐算法，以及文本构建的学术搜索系统。最后分析了模型中仍然存在的问题以及未来研究方向。

# 理论基础

## 图神经网络

图神经网络(graph neural network, GNN)是一种基于深度学习的用于图结构数据的表示学习方法，最早由Scarselli等人[1]在2008年提出。给定有向图，GNN的目标是学习一个函数，将图中的顶点映射到低维向量空间，同时保留图的拓扑结构信息。这些嵌入向量可用于多种下游任务，如顶点分类、连接预测、推荐等。

### 消息传递

GNN的核心思想是消息传递(message passing)[4]：通过聚集邻居信息来更新顶点特征。对于一条有向边，顶点*u*称为起点或源顶点(source node)，顶点*v*称为终点或目标顶点(destination node)，*u*称为*v*的邻居。令表示顶点*v*在第*l*层的特征，消息传递模型定义如下：



(2.1)

其中*φ*是定义在边上的消息函数(message function)，通过组合起点和终点的特征来产生消息；*ρ*是归约函数(reduce function)，用于聚集来自邻居的消息，*N*(*v*)表示顶点*v*的邻居集合；*ψ*是定义在顶点上的更新函数(update function)，通过组合顶点的原始特征和聚集的邻居信息来更新顶点的特征。消息传递过程如图 3所示。顶点的输入特征一般由数据集给定，最后一层的输出特征即顶点的最终表示，可用于下游任务。

(a) 消息函数 (b) 归约函数

图 3 消息传递过程

公式(2.1)描述了消息传递的一次迭代，对应一个GNN层。经过一次消息传递，目标顶点可以聚集一跳邻居的信息。将这一过程重复*n*次，即堆叠*n*个GNN层，则目标顶点能够聚集至多*n*跳邻居的信息，从而捕获到顶点之间的高阶邻接关系。但GNN层数过多会导致过平滑问题[46]，因此实际的GNN模型层数一般设置为2~3层。

以GCN为例，其消息传递过程为，写成向量形式为，即简单的邻居特征求和。对应到公式(2.1)，其消息函数和归约函数分别为：



(2.2)

### 异构图

真实世界的网络通常具有多种类型的顶点和边，例如学术网络（学者、论文、期刊、领域），社交网络（用户、位置、类别）以及生物医学网络（基因、蛋白质、疾病）。因此异构图(heterogeneous graph)被提出并广泛用于多种场景。

异构图定义为一个有向图，以及顶点类型映射和边类型映射，其中*A*和*R*分别表示顶点类型集合和边类型集合，且。二分图(bipartite)就是只有两种顶点类型和一种边类型的特殊的异构图。

异构图复杂的结构使得异构图的表示学习更加困难。大多数现有的异构图GNN模型的第一步是将异构图转换为同构图或二分图，之后使用同构图GNN进行表示学习（同构图GNN同样可用于二分图），最后组合子模块输出的顶点表示得到最终的顶点表示。下面首先介绍一些基本概念的定义，之后对两种常见的转换方法进行解释。

**关系(relation)** 边对应的关系定义为三元组（可简写为*ψ*(*e*)），逆关系表示为。关系和边类型是一一对应的。

**关系二分图(relational bipartite)** 给定异构图*G*和一个关系*r*，图*G*基于关系*r*的二分图*Gr*定义为由关系*r*对应类型的所有边构成的图。

**元路径(metapath)** 元路径*P*定义为具有以下形式的路径：（简写为），其中。元路径描述了顶点类型*A*1和*Al*之间的一个复合关系，该复合关系表达了特定的语义。

**元路径实例(metapath instance)** 给定一条元路径*P*，*P*的实例定义为具有以下形式的路径：，其中。

**基于元路径的邻居(metapath-based neighbor)** 给定一条元路径*P*，顶点*v*基于元路径*P*的邻居定义为通过元路径*P*的实例与顶点*v*连接的顶点集合。如果*P*是对称的则。

**基于元路径的邻居图(metapath-based neighbor graph)** 给定异构图*G*的一条元路径*P*，图*G*基于元路径*P*的邻居图*GP*定义为由所有基于元路径*P*的邻居对构成的图。如果*P*是对称的则*GP*是同构图，否则*GP*是包含*A*1和*Al*两种顶点类型的二分图。

考虑图 4 (a)所示的异构学术网络示例，该异构图包含学者(A)、论文(P)和领域(F)三种类型的顶点以及“写作”(AP)和“所属领域”(PF)两种类型的边。图 4 (b)是元路径PAP及其一个实例p1a1p2，论文p1和p2就是基于元路径PAP的邻居。元路径PAP表达的语义是“同一个学者写作的两篇论文”。图 4 (c)是基于元路径PAP的邻居图*G*PAP，由于元路径PAP是对称的，因此*G*PAP是只包含论文顶点的同构图。图 4 (d)是基于关系AP的二分图*G*AP，即只包含原图中学者和论文顶点及其之间的边。

(a) 异构图 (b) 元路径PAP及实例

(c) 基于元路径PAP的邻居图 (d) 关系AP的二分图

图 4 异构学术网络示例

现有的GNN模型主要使用两种将异构图转换为同构图的方法：基于元路径的邻居图和关系二分图。前者通过预定义一些元路径（例如PAP, PFP），将异构图转换为多个基于元路径的邻居图（例如*G*PAP, *G*PFP）；后者根据目标顶点关联的关系（边类型）将异构图转换为多个关系二分图（例如目标顶点是论文，则转换为*G*AP, *G*FP）。基于元路径的方法需要为每个数据集人工定义一组元路径，这需要专业领域知识。另外，随着图规模的增大，基于元路径的邻居图的边数量会变得非常多（只有几千个顶点、几万条边的异构图转换为基于元路径的邻居图就有几百万条边），因此这种方法无法扩展到大规模数据集。基于关系二分图的转换方法只是将异构图拆分成了多个部分，因此是可扩展的，并且关系集合来自原始图结构，不需要人工定义。基于关系二分图的转换方法目前被大多数异构图GNN模型所采用。

### 小批量训练和邻居采样

在实际训练GNN模型时，当图的规模过大时无法进行全图训练，因为在全图上进行消息传递会消耗大量的计算资源，可能导致内存溢出。为此，需要使用小批量训练(mini-batch training)和邻居采样(neighbor sampling)技术。小批量训练方法将顶点集合划分为多个批次(batch)，每次只计算一个批次的顶点的输出，这一个批次的顶点称为目标顶点或种子顶点(seed node)。使用邻居采样方法进行多层消息传递的核心思想是：如果要计算某个顶点第*L*层的表示，则需要计算这个顶点的邻居第*L*-1层的表示，又需要计算这些邻居的邻居第*L*-2层的表示……这一过程一直持续到输入层。

(a) 原图 (b) 输出层

(c) 第2层消息传递 (d) 第1层消息传递

图 5 两层消息传递邻居采样示例

图 5给出了一个使用邻居采样方法进行两层消息传递的示例。图 5 (a)是完整的图。假设当前批次的目标顶点是顶点8（红色顶点），如图 5 (b)所示。要计算目标顶点第2层的表示，根据消息传递规则，需要聚集计算一阶邻居（黄色顶点）第1层的表示，如图 5 (c)所示；从而需要聚集二阶邻居（绿色顶点）的输入特征，如图 5 (d)所示。

综上所述，在小批量训练中，每个批次的*n*层消息传递只需要*n*阶邻居的输入特征，以及每一层消息传递对应的子图，即可计算目标顶点的输出特征。图 5 (d)和图 5 (c)中有颜色的顶点和边恰好分别构成了第1层和第2层消息传递所需要的子图，这种由消息传递的输入顶点和输出顶点构成的二分子图称为消息流图(message flow graph, MFG)。小批量训练使得GNN模型能够适配任意规模的数据集。另外，在真实世界的图数据中，一个顶点的邻居数量可能非常多，此时可以使用邻居采样，为每个顶点采样固定数量的邻居（而不是使用全部邻居）进行消息传递，从而提高计算效率。

### 下游任务

GNN输出的顶点表示向量可用作下游任务的输入，常见的下游任务有以下几种：

（1）顶点分类

顶点分类是图表示学习最常见的下游任务，属于半监督学习。在顶点分类任务中，每个顶点*v*有一个标签，将顶点嵌入输入下游分类器（例如逻辑回归、单层全连接网络等）计算出每个顶点的预测概率，之后通过最小化交叉熵损失来更新模型参数：



(2.3)

常用的评价指标包括准确率(accuracy)、Macro-F1和Mirco-F1。对于多分类问题，准确率和Micro-F1相等。

（2）连接预测

连接预测任务旨在训练一个二分类器，预测两个顶点之间是否存在一条边，属于无监督学习。在连接预测任务中，使用两个顶点嵌入的内积的sigmoid值作为这两个顶点之间存在一条边的概率，即。通过负采样(negative sampling)得到负样本边，通过最小化以下损失函数来更新模型参数：



(2.4)

其中*E*+是观测到的边（正样本边）集合，*E*-是从未观测到的边集合（*E*+的补集）中采样出的负样本边集合，*σ*是sigmoid函数。常用的评价指标包括ROC曲线下面积(area under the ROC curve, AUC)和平均精度(average precision, AP)。

（3）顶点聚类

顶点聚类任务将顶点嵌入输入K-Means算法，类簇个数设置为标签类别数。顶点聚类和前两个任务最大的区别在于聚类步骤和GNN模型的训练过程是独立的。常用的评价指标包括归一化互信息(normalized mutual information, NMI)和调整的兰特指数(adjusted rand index, ARI)。

## 自监督学习和对比学习

### 自监督学习

深度神经网络在有监督学习的多种机器学习任务上展现出优越的性能，例如计算机视觉（图像分类、语义分割），自然语言处理（预训练语言模型、语义分析、问答）和图学习（顶点分类、图分类）。然而有监督学习严重依赖于昂贵的人工标注，并且泛化错误、虚假相关性和对抗性攻击等问题[48]。作为一种替代方案，自监督学习(self-supervised learning)因其数据效率和泛化能力而受到广泛关注。

“自监督学习”一词最早出现在机器人技术中，通过利用不同输入传感器信号之间的关系自动标注训练数据，之后机器学习社区进一步发展了这个想法。自监督学习的特征可总结为：（1）通过半自动化过程从数据本身获取“标签”；（2）使用其他部分数据预测一部分数据。其中“其他部分”可能不完整、变形、扭曲或损坏（即数据增强技术）。

自监督学习可以被视为无监督学习的一个分支，因为不涉及手动标签。二者的主要区别是，无监督学习专注于发现特定的数据模式（例如聚类、社区发现或异常检测），而自监督学习旨在恢复，而这属于有监督设定。

自监督学习成功的最关键点是它找到了一种方法来利用大量无标注数据。自监督学习的直觉是利用数据固有的共现关系作为自监督，这种方法是通用的。自监督方法主要分为三类：（1）生成式(generative)：训练一个编码器将输入*x*编码为显式向量*z*，以及一个解码器从*z*中重建*x*；（2）对比式(contrastive)：训练一个编码器将输入*x*编码为显式向量*z*，并评估相似性；（3）对抗式(adversarial)：训练一个生成器（包括编码器和解码器）来生成假样本，以及一个判别器来将其与真样本区分开（例如GAN）。

### 对比学习

对比学习(contrastive learning)是自监督学习的一个重要分支。从统计的角度，机器学习模型分为生成式和判别式模型。给定输入*X*和目标*Y*的联合分布，生成式模型计算，而判别式模型试图建模。因为大部分表示学习任务希望对输入之间的关系进行建模，长期以来，人们认为生成式模型是表示学习的唯一选择。然而，对比学习的最新突破（例如Deep InfoMax, MoCo和SimCLR）揭示了判别式模型在表征方面的潜力。对比学习旨在通过以下噪声对比估计(noise contrastive estimation, NCE)目标函数来学习比较样本之间的相似性：



(2.5)

其中*x*+是*x*的正样本，*x*-是*x*的负样本，*f*是一个编码器（表示函数）。相似性度量和编码器可能因任务而异，但框架是相同的。

自监督学习和对比学习已被成功应用于计算机视觉和自然语言处理领域。最近，自监督学习在图表示学习领域的应用吸引了大量的研究兴趣。和其他领域一样，图结构数据的标签通常难以获取，而自监督学习可以从图结构本身获取监督信号，从而能够捕获图的固有结构信息。

自监督学习的核心问题在于为无标注数据定义适当的目标函数，即“学习什么”。Peng, Zhen等人在2020年提出的自监督图表示学习框架S2GRL[49]通过预测图中一对顶点的相对上下文位置（即预测最短跳步数）进行自监督学习。Wang, Ping等人在2021年提出的上下文嵌入自监督学习框架SLiCE[50]将使用来自整个图的全局信息的静态表示学习方法与学习异构网络中上下文顶点表示的局部注意力机制结合起来，通过自监督学习来学习高阶交互关系，从而学习一个用于上下文翻译的运算符。Wang Xiao等人在2021年提出了使用跨视图对比学习的自监督异构图神经网络HeCo[22]，该模型分别使用网络结构编码器和元路径编码器捕获图的局部结构和高阶结构，通过这两个视图学习到的顶点嵌入进行对比学习。

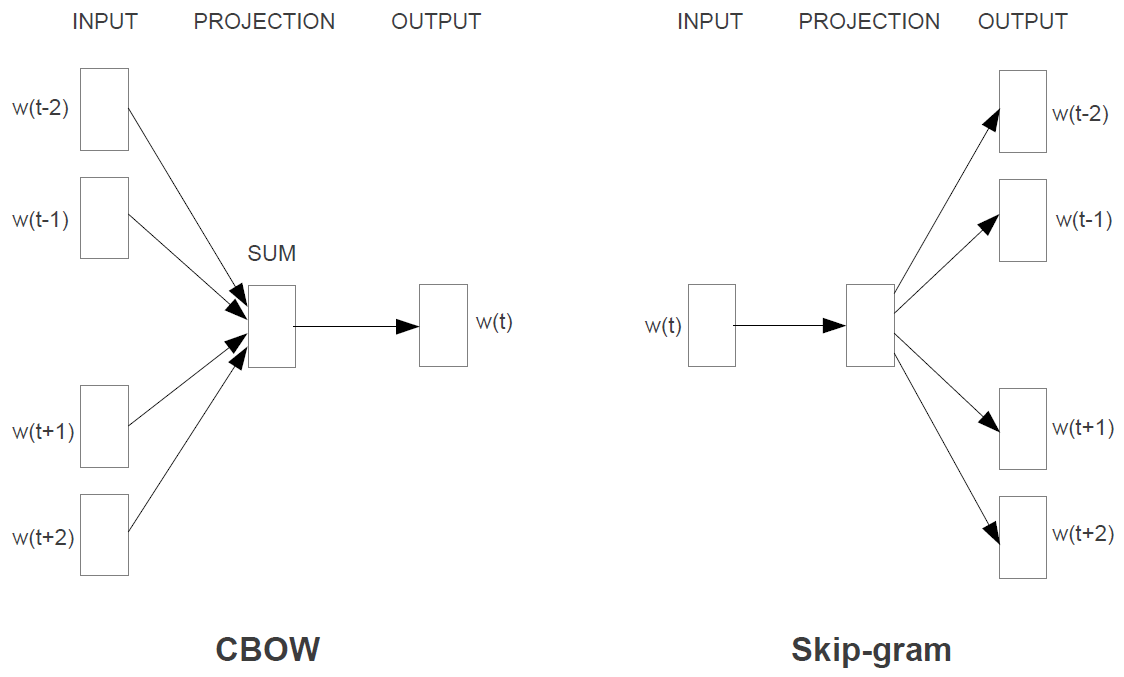
## 词向量模型

### Skip-gram模型

在自然语言处理中，最简单的词向量表示方法是one-hot编码，使用长度为整个词汇表大小的向量来表示单词。对于词汇表中的每一个单词，向量对应位置的元素为1，其他元素均为0。One-hot编码主要有两个缺陷：（1）向量的长度会随词汇表的增大而增大，常用的词汇表一般都达到十万甚至百万数量级，使用one-hot编码将造成“维数灾难”；（2）任意两个单词的向量都是相互正交的，在欧式空间中的距离都是，无法表达单词包含的语义信息。

为了解决one-hot编码的不足，Hinton等人[47]于1988年提出了分布式表示(distributed representation)。其核心思想是通过训练将词汇表中的每一个单词映射到一个固定长度的向量，所有词向量构成一个词向量空间，每一个词向量可视为该空间中的一个点，在该空间中可以使用欧氏距离、余弦相似度等方式通过单词之间的距离判断语义相似性。

Word2vec是Tomas Mikolov等人[23]于2013年提出的一种用于从大规模语料库中通过训练获得高质量词向量的方法。word2vec有两种训练模型，分别是CBOW (Continuous Bag-of-Words)模型和Skip-gram模型，模型结构如图 6所示。



(a) CBOW (b) Skip-gram

图 6 CBOW和Skip-gram模型结构

这两个模型都包含3层：输入层、隐藏层和输出层。前者的训练目标是由上下文预测中心词*wt*，后者是由中心词预测上下文。以Skip-gram模型为例，给定一系列训练单词，目标是最大化平均对数概率：



(.6)

其中*c*是上下文大小。Skip-gram模型使用softmax函数定义：



(.7)

其中*vw*和*vw’*分别是单词*w*的输入和输出向量表示，*W*是词汇表大小。由于*W*通常会很大（105~107数量级），计算的复杂度与*W*成正比，直接使用公式(2.7)计算是不现实的。因此word2vec使用了层次softmax和负采样两种方法来提高训练速度，本文不再详细介绍。

训练得到的词向量可以捕获到单词的语法和语义规律：具有相同关系的两对单词的词向量应当具有相似的偏移量，如图 7所示。利用这一特性，可以使用简单的向量代数运算来解决语法和语义推理问题，已在自然语言处理的各种任务中得到广泛应用。例如：

vec(king) - vec(man) + vec(woman) ≈ vec(queen)

vec(China) - vec(Beijing) + vec(France) ≈ vec(Paris)



图 7 词向量包含的语义规律

### 在图表示学习领域的应用

Skip-gram是一种语言模型，DeepWalk首次将其应用到图表示学习领域。该论文认为，图中的顶点在短随机游走中出现的频率与自然语言中的单词频率服从相似的分布，因此用于建模语言的技术也可用于建模网络结构。DeepWalk的核心思想是在图中进行给定数量和长度的随机游走，之后将顶点视为单词、顶点序列视为句子，使用Skip-gram模型学习顶点的表示向量，如图 8所示。metapath2vec在DeepWalk的基础上采用基于元路径的随机游走策略采样异构图的顶点序列，从而能够进行异构图表示学习。使用这种方法得到的顶点表示在下游任务中的性能虽然并不如GNN模型好，但可以作为异构图数据集的顶点输入特征。



图 8 DeepWalk算法

## 小结

本章首先介绍了图神经网络的基本概念、其核心思想消息传递模型、异构图相关的一些基本概念、为了适配大规模数据集所采用的小批量训练和邻居采样方法以及常见的下游任务。接着介绍了自监督学习和对比学习的基本思想及其与图表示学习领域的结合。最后介绍了词向量模型Skip-gram及其在图表示学习领域的应用。

# 基于对比学习的异构图神经网络模型

## 引言

目前大部分异构图神经网络模型属于半监督学习范式，然而在某些真实世界环境下获取标签是具有挑战性的且昂贵的。对比学习旨在通过提取数据中的正负样本，最大化目标样本与正样本之间的相似度，同时最小化目标样本与负样本之间的相似度。通过这种方式，即使没有标签也能够学习到有区分度的嵌入。对比学习在计算机视觉和自然语言处理领域已被广泛使用，但在异构图表示学习领域的研究还相对较少。

设计使用对比学习的异构图神经网络需要解决以下三个基本问题：

（1）**如何设计异构对比机制。**异构图具有复杂的结构，元路径是一种常用的用于捕获异构图高阶关系的方法，不同的元路径表达了不同的语义。然而仅在元路径视图上进行对比学习是不够的，因此需要研究异构跨视图对比机制。

（2）**如何在异构图中选择适当的视图。**选择的视图应当覆盖局部和高阶结构，例如网络结构(schema)视图自然地捕获了局部结构，元路径视图可用于提取高阶结构。

（3）**如何设置一个困难的对比任务。**适当的对比任务能够帮助模型学习到更具有区分度的嵌入，如果两个视图太相似则无法学习到有价值的嵌入。

HeCo模型[22]分别使用网络结构编码器和元路径编码器捕获图的局部结构和高阶结构，通过这两个视图学习到的顶点嵌入计算目标顶点及其正样本之间的相似度作为对比损失。该模型首次将对比学习方法应用到异构图表示学习领域，但存在几个不足：

（1）**可扩展性差。**HeCo模型的元路径编码器使用不同的元路径将异构图转换为基于元路径的邻居图；另外，顶点正样本的选择策略是基于两个顶点之间的元路径实例数量。如2.1.2节所述，基于元路径的方法无法扩展到大规模数据集，因此当图的规模较大时该模型将无法使用。

（2）**网络结构简单。**HeCo模型的网络结构编码器是单层GAT，元路径编码器是单层GCN。这种简单的结构使得模型的学习能力有限，当数据规模较大时模型性能会显著下降。

（3）**输入特征简单。**HeCo模型使用的异构图数据集至多只有一种类型的顶点有输入特征，而对于没有输入特征的顶点类型使用one-hot编码，这种编码不包含任何与顶点相关的语义信息，并且当顶点数量较多时会导致“维数灾难”。

## 基于对比学习的关系感知异构图神经网络

异构图表示学习问题的形式化定义如下。给定异构图，以及顶点类型映射和边类型映射，其中*A*和*R*分别表示顶点类型集合和边类型集合，且，模型的目标是学习一个函数，将图中的顶点映射到低维向量空间，同时保留图的拓扑结构信息。

本文在HeCo模型的基础上进行了改进，提出了基于对比学习的关系感知异构图神经网络(Relation-aware Heterogeneous Graph Neural Network with Contrastive Learning, RHCO)。RHCO模型由输入特征转换、网络结构编码器、正样本图编码器和损失函数四部分组成，模型结构如图 9所示。



图 9 RHCO模型结构

网络结构编码器和正样本图编码器分别用于编码图的局部结构和高阶结构，从而充分利用了异构图的结构信息。使用这两个编码器输出的不同视图下的顶点嵌入进行对比学习，计算对比损失；网络结构编码器输出的顶点嵌入经过C&S[20]标签平滑处理后输入下游分类器，得到分类损失。最后将对比损失和分类损失加权求和得到最终的损失函数。

在异构图中，一个目标顶点通常关联了多种关系，目标顶点在不同关系中扮演了不同的角色，而现有的异构图表示学习方法没有显式地利用关系的角色。因此，本文提出的表示学习方法考虑了关系信息，每个顶点都关联了多个特定于关系的表示，即顶点表示是关系感知的，从而反映顶点相对于不同关系的特性。

### 输入特征转换

由于异构图中不同类型顶点的输入特征往往具有不同的维数、属于不同的隐含空间，因此通过类型相关的线性变换将所有类型顶点的输入特征转换为相同的维数，并映射到相同的隐含向量空间，如图 9 (a)所示。首先，对于顶点*v*及其关联的关系*r*，分别通过顶点类型和关系类型相关的线性变换将其表示映射到相同的隐含空间：



(.1)

其中和分别是顶点*v*关于关系*r*的第*l*层表示和关系*r*本身的第*l*层表示，设置为顶点*v*的输入特征*xv*，使用one-hot编码，和是类型相关的转换矩阵。

### 网络结构编码器

网络结构编码器旨在学习目标顶点在网络结构视图下的嵌入，从而能够捕获网络的局部结构信息，如图 9 (b)所示。

假设目标顶点*v*关联的关系集合为*Rv*，在关系*r*下的邻居集合记为。对于目标顶点*v*，不同类型的邻居、同种类型的不同邻居对其嵌入有不同的贡献，因此使用注意力机制分别聚集同类型不同邻居的信息和不同类型的邻居信息。

首先使用顶点层次的注意力聚集关系*r*下的邻居信息：



(.2)

其中表示关系*r*下的邻居*u*对目标顶点*v*的重要性，||表示向量拼接，*σ*是激活函数。

公式(3.2)的本质是在关系*r*对应的关系二分图*Gr*上进行消息传递。这里使用了GAT的基本思想，但使用关系的表示向量作为注意力权重向量，因此目标顶点的嵌入包含了关系信息，即该嵌入是关系感知的。

为了增强学习过程的稳定性，本文将公式(3.2)扩展到多头注意力机制，即将这一过程独立地重复*K*次，之后将*K*个注意力头的输出向量拼接起来：



(.3)

公式(3.1)和(3.2)描述了一个GNN层的消息传递过程，将其堆叠*L*层，可以得到目标顶点*v*关于所有关系类型的嵌入。之后，使用类型层次的注意力将这些嵌入组合起来得到网络结构编码器的输出，即目标顶点*v*在网络结构视图下的嵌入：



(.4)

其中*βv*,*r*表示关系*r*对目标顶点*v*的重要性，*Ur*和*Vr*分别为顶点表示和关系表示的转换矩阵。

由于公式(3.4)仅考虑了一阶邻居关系集合*Rv*，因此目标顶点*v*在网络结构视图下的嵌入捕获了异构图的局部结构信息。

### 正样本选择策略

为了解决3.1节所述的可扩展性问题，本文使用预训练的两层HGT模型计算出的注意力权重为每个目标顶点选择固定数量的同类型顶点作为正样本。对于目标顶点*v*和通过*φ*类型的中间顶点连接的同类型顶点*u*，*u*对*v*的注意力权重使用下式计算：



(.5)

其中*eu*,*v*表示由预训练的HGT计算出的顶点*u*对顶点*v*的注意力权重。对于目标顶点*v*，选择注意力权重最大的*T*pos个同类型顶点作为正样本，所有正样本对构成的图即为正样本图，如图 10所示。

正样本图是目标类型的顶点构成的同构图，其中的边包含了元路径信息，因为公式(3.5)中的中间顶点*u’*是目标顶点*v*的*φ*类型一阶邻居，而顶点*u*是*v*的同类型二阶邻居，因此顶点*u*本质上是顶点*v*的基于元路径的邻居，从而通过*φ*类型的中间顶点连接的正样本对构成的正样本图本质上是基于元路径*p*的邻居图*Gp*。例如，图 10中目标顶点是论文(P)，则通过学者(A)顶点与目标顶点连接的论文顶点是基于元路径PAP的邻居。



图 10 正样本选择策略

对于目标顶点关联的所有邻居类型*Φv*分别构造正样本图，用于下一节所述的正样本图编码器。另外，使用下式计算出一个整体的注意力权重：



(.6)

之后按同样的方式构造一个正样本图*G*pos。该正样本图综合考虑了所有元路径，用于后续的对比损失计算及标签平滑。

与HeCo模型所采用的基于元路径的正样本选择策略相比，本文提出的正样本选择策略虽然本质上也是选择基于元路径的邻居作为正样本，但这种方法不需要显式地构造基于元路径的邻居图，而是以小批量训练的方式逐个批次计算目标顶点的正样本，因此可以在大规模数据集上使用。

### 正样本图编码器

正样本图编码器旨在学习目标顶点在正样本视图下的嵌入，从而能够捕获网络的高阶结构信息，如图 9 (c)所示。

假设目标顶点*v*关联的邻居类型集合为*Φv*，对应的元路径集合为。顶点*v*基于元路径的邻居集合就是在对应的正样本图*Gp*中的邻居集合，也是顶点*v*的正样本集合。对于元路径*p*，使用对应正样本图*Gp*上的单层GCN来聚集邻居（正样本）特征：



(.7)

其中*xu*是顶点*u*的输入特征，*Wp*是转换矩阵。

得到目标顶点*v*关于所有元路径的嵌入后，使用语义层次的注意力将这些嵌入组合起来得到正样本图编码器的输出，即目标顶点在正样本视图下的嵌入：



(.8)

其中*βp*表示元路径*p*对目标顶点*v*的重要性，*Wpg*和*bpg*是可学习参数，*q*是语义层次的注意力向量。

如上所述，正样本对本质上是基于元路径的邻居，因此目标顶点*v*在正样本视图下的嵌入捕获了网络的高阶结构信息。

### 损失函数

得到以上两个视图下的顶点嵌入和后，首先使用两层全连接网络对其进行线性变换，之后使用下式计算对比损失：



(.9)

其中*Nv*为目标顶点*v*在正样本图*G*pos中的邻居，也是其正样本集合，其他顶点视为顶点*v*的负样本，*τ*为对比损失的温度参数，*λ*为对比损失平衡系数。当使用小批量训练时，公式(3.9)中的*V*表示一个批次的目标顶点集合。注意到计算对比损失不仅需要目标顶点的嵌入，还需要其正样本的嵌入，因此在计算每个批次的消息传递时，实际目标顶点集合应为。

对比损失用于在无标签的场景下仅利用数据本身进行学习。对于有标签的顶点分类任务，利用标签信息将有助于提升下游任务的性能。本文将网络结构编码器输出的顶点嵌入在正样本图上进行标签平滑处理后输入下游分类器。标签平滑是指对于有标签顶点（训练集）直接使用真实标签，对于无标签顶点（测试集）使用模型输出的嵌入，即：



(.10)

从而得到基础预测。之后按下式进行多次迭代：



(.11)

其中是标签传播图的归一化邻接矩阵，本文使用3.2.3节所述的正样本图*G*pos作为标签传播图。经过*T*次迭代后得到最终预测，之后使用公式(2.3)计算分类损失。

整体的损失函数如下：



(.12)

其中*α*为对比损失权重系数。

### 模型分析

与原HeCo模型相比，本文提出的RHCO模型进行了以下几点改进：

（1）网络结构编码器中的注意力向量改为关系的表示向量，使得输出的嵌入能够包含关系信息；

（2）正样本选择方式由元路径实例数量改为预训练的HGT计算的注意力权重；

（3）元路径视图编码器改为正样本图编码器，不需要显式地构造基于元路径的邻居图，使得模型能够适配小批量训练，解决了无法扩展到大规模数据集的问题；

（4）损失函数增加分类损失，训练方式由无监督改为半监督，并在最后增加了标签平滑处理步骤，有助于提升下游任务的性能。

## 实验设计与分析

### 数据集

本文使用以下三个真实世界的异构学术网络数据集进行顶点分类实验，包括一个小规模数据集和两个大规模数据集，具体数据如表 1所示。

表 1 异构图表示学习数据集

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据集 | 顶点 | 边 | 目标顶点 | 类别数 |
| DBLP | 学者(A): 4057  论文(P): 14328  会议(C): 20  关键词(T): 7723 | P-A: 19645  P-C: 14328  P-T: 85810 | 学者 | 4 |
| ogbn-mag | 学者(A): 1134649  论文(P): 736389  领域(F): 59965  机构(I): 8740 | P-A: 7145660  P-P: 5416271  P-F: 7505078  A-I: 1043998 | 论文 | 349 |
| oag-venue | 学者(A): 2248205 | P-A: 6349317 | 论文 | 360 |
| 论文(P): 1852225 | P-P: 9194781 |
| 领域(F): 120992 | P-F: 17250107 |
| 机构(I): 13747 | A-I: 1726212 |

**DBLP[[4]](#footnote-5)** DBLP论文数据集，包含学者(A)、论文(P)、会议(C)、关键词(T)四种顶点。学者和论文顶点的输入特征是关键词的词袋表示，会议和关键词顶点的输入特征使用one-hot编码。目标顶点是学者，按研究领域分为数据库、数据挖掘、机器学习、信息检索四个类别。采用随机划分，训练集、验证集、测试集的大小分别为240、1000、1000。

**ogbn-mag[[5]](#footnote-6)** Open Graph Benchmark (OGB)是斯坦福大学公开的一组真实世界的大规模基准图数据集，提供了统一的数据加载和性能评估方式，被越来越多的GNN研究者采用。其中ogbn-mag数据集是使用微软学术数据构造的一个异构学术网络，包含学者(A)、论文(P)、领域(F)、机构(I)四种顶点。论文的输入特征是预训练的128维word2vec词向量，其他类型的顶点没有输入特征。本文使用2.3.2节所述的方法，使用metapath2vec预训练的顶点嵌入作为输入特征。预测任务是预测论文所属期刊，共有349个类别。采用数据集提供的划分（按年份），训练集、验证集、测试集的大小分别为629571、64879、41939。

**oag-venue** oag-venue数据集在4.3.1节所述的oag-cs数据集的基础上删除了期刊顶点，并将论文数大于等于1000的期刊作为与其关联的论文的标签。图中包含学者(A)、论文(P)、领域(F)、机构(I)四种顶点。论文和领域顶点的输入特征是微调后的SciBERT模型输出的128维向量，其他类型的顶点使用metapath2vec预训练的顶点嵌入作为输入特征。预测任务是预测论文所属的期刊，共有360个类别。按年份划分，2014年及以前、2015~2017年、2018年及以后的论文分别作为训练集、验证集和测试集，大小分别为402457、280762和255387。

### 对比模型

本文将RHCO模型与目前最新的异构图神经网络模型进行对比，包括R-GCN[11], HGT[17], HGConv[18], R-HGNN[19], C&S[20]和HeCo[22]。

与R-GCN, HGT和HGConv相比，RHCO在顶点嵌入中考虑了关系信息，学习了关系感知的顶点表示，从而能够区分同一个顶点在不同关系中的不同角色，有助于学习到更好的顶点嵌入；与R-HGNN相比，RHCO增加了正样本图编码器，并在最终的损失函数中增加了对比损失，从而能够在不同视图下学习顶点嵌入，充分利用异构图的结构信息；与HeCo相比，RHCO改进了网络结构编码器，并通过替换正样本选择策略解决了可扩展性问题。

### 实现细节

本文使用PyTorch[[6]](#footnote-7)和DGL[[7]](#footnote-8)框架实现RHCO模型和所有的对比模型。对于RHCO模型，顶点嵌入维数*d* = 64，关系嵌入维数*d*rel = 8，注意力头数*K* = 8，网络结构编码器的层数*L* = 2，每个顶点的正样本个数*T*pos = 5，dropout概率为0.5，温度参数*τ* = 0.8，对比损失平衡系数*λ* = 0.5，对比损失权重*α* = 0.9。标签平滑中的*γ* = 0.5，迭代次数*T* = 50。对比模型使用原论文中的参数。

对于ogbn-mag和oag-venue数据集使用小批量训练和邻居采样，批大小为512，邻居采样数为10，迭代150次(epoch)；对于DBLP数据集使用全图训练，迭代10次。使用Adam优化器，学习率为0.001。

### 评价指标

本文使用准确率(accuracy)和Macro-F1作为顶点分类任务的评价指标：



(.13)

其中*C*为类别数，*Pi*和*Ri*分别为第*i*个类别的准确率和召回率。

### 实验结果

本文在顶点分类任务上评价模型性能，将模型学习到的顶点嵌入输入线性分类器（单层全连接网络）并预测顶点类别。各模型在三个数据集上的评价指标如表 2所示。

表 2 顶点分类实验结果

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 数据集 | 评价指标 | R-GCN | HGT | HGConv | R-HGNN | C&S | HeCo | RHCO |
| DBLP | 准确率 | **0.9490** | 0.7860 | 0.9060 | 0.8680 | 0.7970 | 0.9070 | 0.8840 |
| Macro-F1 | **0.9433** | 0.7837 | 0.8951 | 0.8591 | 0.7799 | 0.9032 | 0.8732 |
| ogbn-mag | 准确率 | 0.3720 | 0.4497 | 0.4851 | 0.5201 | 0.3558 | 0.3043 | **0.5662** |
| Macro-F1 | 0.1970 | 0.2853 | 0.3148 | 0.3164 | 0.1863 | 0.0985 | **0.3433** |
| oag-venue | 准确率 | 0.1577 | 0.8359 | 0.8144 | 0.9615 | 0.1392 | 0.1361 | **0.9623** |
| Macro-F1 | 0.1088 | 0.7628 | 0.7486 | 0.9057 | 0.0878 | 0.0681 | **0.9186** |
| 平均值 | 准确率 | 0.4929 | 0.6905 | 0.7352 | 0.7832 | 0.4307 | 0.4491 | **0.8042** |
| Macro-F1 | 0.4164 | 0.6106 | 0.6528 | 0.6937 | 0.3513 | 0.3566 | **0.7117** |

从结果中可以看出，在ogbn-mag和oag-venue数据集上，RHCO均超越了所有对比模型，并且在三个数据集上的平均准确率和Macro-F1与最好的对比模型相比分别取得了2.1%和1.8%的提升，这表明在大规模、结构复杂的数据集上RHCO能够学习到更好的顶点嵌入。

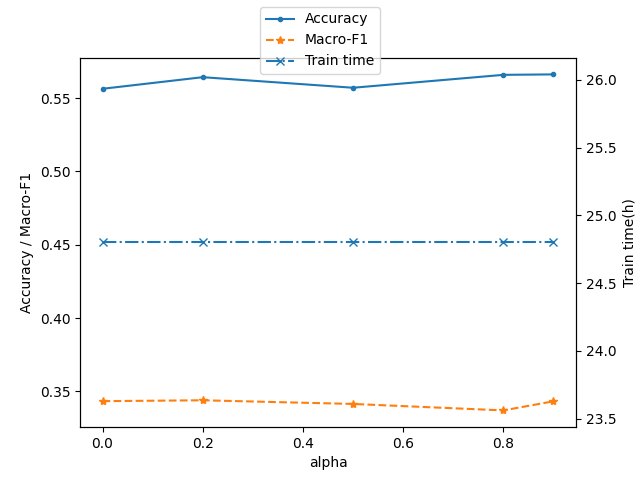
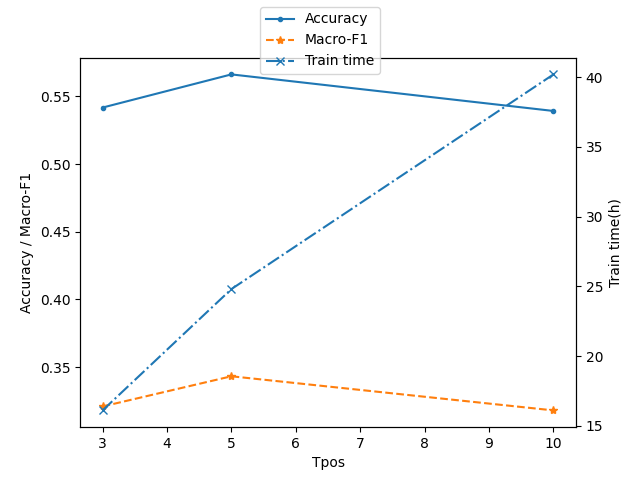
在DBLP数据集上，R-GCN取得了最好结果，甚至超越了其他更复杂的异构图神经网络模型。主要原因是DBLP数据集的目标顶点是学者，而学者顶点只有论文一种类型的邻居，因此在消息传递中能够获得的信息较少，这导致RHCO网络结构编码器中类型层次的注意力和正样本图编码器中语义层次的注意力实际上并没有作用。因此对于规模小、结构简单的数据集，R-GCN采用的简单邻居平均比复杂的消息传递更有效；而在另外两个大规模数据集上，R-GCN会发生严重的过拟合，训练集准确率可以达到90%以上，但测试集准确率则远低于其他模型。因此对于大规模、信息更加丰富的异构图，需要更复杂的异构图神经网络模型来挖掘其中的信息。

由于原始的HeCo无法应用于大规模数据集，因此本文在两个大规模数据集的实验中，将其元路径编码器改为RHCO的正样本图编码器。实验结果表明，仅依靠完全无监督的对比损失难以学习到对下游任务有效的顶点嵌入。

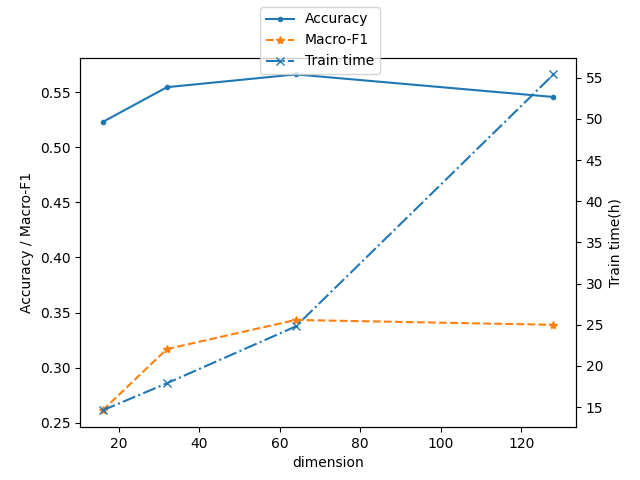
对于C&S模型，其作者认为该模型主要的创新点在于“简单线性模型与标签传播相结合即可以达到与GNN相近的性能”。然而本文的实验结果表明，这一结论仅在小规模数据集上成立。在大规模数据集上，简单线性模型（如MLP）的性能很差，而标签传播提升的性能也非常有限。

### 参数敏感性分析

本文在ogbn-mag数据集上分析RHCO模型的三个主要参数对模型的性能影响，包括对比损失权重*α*、正样本个数*T*pos和顶点嵌入维数*d*。其中*α*的取值范围为[0, 0.2, 0.5, 0.8, 0.9]，*T*pos的取值范围为[3, 5, 10]，*d*的取值范围为[16, 32, 64, 128]。参数的不同取值对模型性能和训练时间的影响如图 11所示。

(a) 对比损失权重 (b) 正样本个数



(c) 顶点嵌入维数

图 11 RHCO模型参数敏感性分析实验结果

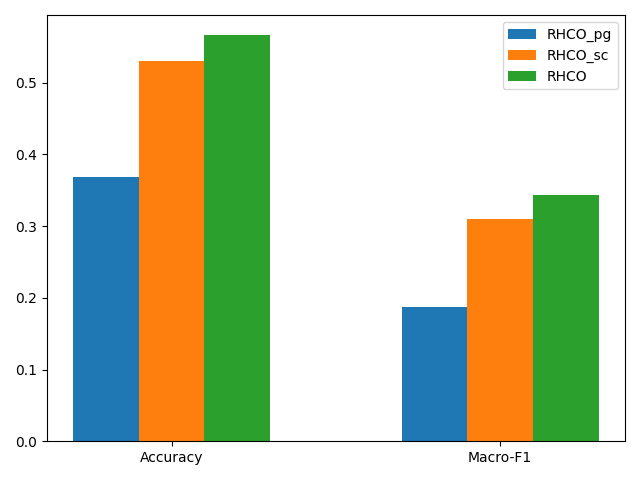
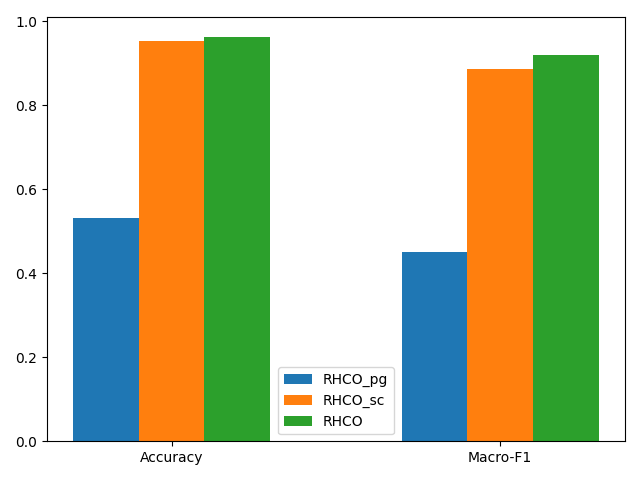
图 11 (a)显示对比损失权重*α*的变化对模型性能的影响不明显，当*α* = 0.9时达到最好性能，这表明在无监督的对比学习中只需加入很少的监督信息即可对下游任务的性能有较大提升（相比于HeCo模型）。

图 11 (b)展示了正样本个数*T*pos对模型性能的影响。*T*pos决定了公式(3.7)中的邻居个数和公式(3.9)中的正样本集合大小，即。随着正样本个数的增加，模型性能首先会有一定的提升，这是由于在正样本图编码器中，目标顶点能够聚集更多来自正样本的信息，并且对比损失分子上的正样本个数增加能够更好地与负样本进行对比。然而当正样本数量过多时，目标顶点与其正样本之间的标签一致性会下降，从而影响标签传播的性能，导致最终的模型性能下降。

图 11 (c)展示了顶点嵌入维数*d*对模型性能的影响。可以看出，随着嵌入维数的增大，模型性能先提升后下降，在*d* = 64时达到最高，同时训练时间会显著增加。这表明RHCO需要一个合适的嵌入维数来编码顶点信息，维数过小时表达能力不足，而维数过大时会引入额外的冗余信息。

### 消融实验

为了探究RHCO的两个主要组件对模型性能的影响，本文设计了RHCO\_sc和RHCO\_pg两个变体，在ogbn-mag和oag-venue两个大规模数据集上验证网络结构编码器和正样本图编码器对整个模型的作用。其中，RHCO\_sc仅使用网络结构编码器编码顶点嵌入，对比损失中正负样本对应的嵌入也来自网络结构视图；RHCO\_pg仅使用正样本图编码器编码顶点嵌入，对比损失中正负样本对应的嵌入以及用于下游任务的嵌入也来自正样本视图。这两个变体与整体RHCO模型的性能比较如图 12所示。

(a) ogbn-mag数据集 (b) oag-venue数据集

图 12 RHCO模型消融实验结果

从结果中可以看出，整体RHCO模型的性能优于两个变体，这证明了跨视图对比学习的有效性。其次，RHCO\_sc的性能也很好，尤其是在oag-venue数据集上仅与RHCO有很小的差距，这表明聚集不同类型一阶邻居的信息相比聚集同类型高阶邻居的信息能够获得更好的顶点嵌入，因为前者利用了异构图中多种类型的邻接关系，而后者本质上仅考虑了一种顶点类型。最后，RHCO\_pg的性能最差，这表明基于元路径的方法在大规模数据集中并不足以捕获图的异构性。

值得注意的是，本文得出的这一结论与HeCo原论文恰好相反。原作者通过消融实验表明HeCo\_mp的性能比HeCo\_sc更好，从而证明了元路径在异构图表示学习中是一种强大的方法。然而，如2.1.2节中所述，元路径方法并不能应用于大规模数据集。另外，即使本文通过改变正样本选择策略的方式使得元路径方法能够使用，但是通过消融实验发现，其在大规模数据集上的性能仍然很差。因此，虽然元路径在异构图表示学习中是一种被广泛使用的、强大的工具，但其应用范围仅限于小规模数据集。

## 小结

本章介绍了基于对比学习的异构图神经网络模型RHCO。该模型在顶点嵌入中融入了关系信息；修改了正样本选择策略，使得模型能够扩展到大规模数据集；通过标签平滑处理提升了模型性能。在顶点分类任务上的定量实验证明了RHCO模型在异构图表示学习方面的性能提升。

# 基于图神经网络的学术推荐算法

## 问题定义

在传统的推荐系统中，有*m*个用户和*n*个物品，以及用户-物品交互矩阵，表示用户和物品的交互关系（例如点击、购买、评分等）。给定*Y*，目标是预测用户*u*对以前未交互过的物品*v*是否有潜在兴趣，即学习一个预测函数，表示用户*u*与物品*v*产生交互的概率，*Θ*表示模型参数。这种仅利用历史交互信息的推荐方法称为协同过滤(collaborative filtering)。协同过滤是最传统的推荐方法，目前已被广泛应用，但存在数据稀疏性和冷启动问题（无法对新用户或新物品进行推荐）。

为了解决以上问题，研究者尝试将辅助信息(side information)引入推荐，例如用户社交网络和外部知识图谱。由于推荐系统中的数据本质上具有图结构（用户-物品二分图、社交网络或知识图谱），而GNN在捕获顶点之间的连接性以及图表示学习方面非常强大，因此在推荐系统中利用GNN成为一个新的研究趋势。

与大多数现有的推荐系统文献不同，本文研究的推荐问题是领域学者排名问题，即给定一个包含学者、论文、领域等信息的异构学术网络，用户输入一个任意的领域关键词，能够检索出与该领域最相关的学者并排名，如图 2所示。

领域学者排名问题与传统的top-k推荐任务可大致按以下方式类比：领域相当于“用户”，学者相当于“物品”，即推荐任务是给领域推荐学者，学者在某个领域的论文被引用相当于“交互关系”。其中最大的区别在于“用户”不是使用系统的用户，而是用户输入的领域关键词。

基于以上分析，领域学者排名问题与传统的推荐任务相比存在一个困难：用户输入的领域关键词是任意的，而不一定是图中存在的领域顶点。也就是说“用户”集合是一个无穷集合，本文将该问题称为“无穷用户集合问题”。对此，需要一种能够将任意领域关键词编码为向量的方法。

## 算法设计

本文针对领域学者排名问题提出了基于图神经网络的学术推荐算法(Graph Neural Network based Academic Recommendation Algorithm, GARec)，整体框架如图 13所示。GARec算法整体上分为两步：（1）论文召回，将输入的查询词和论文标题分别使用微调后的SciBERT模型编码为向量并计算相似度，取相似度最高的*m*篇论文作为召回结果；（2）精排，使用召回的论文及其关联的学者构造子图，使用第三章提出的RHCO模型学习学者顶点嵌入，并与查询向量计算相似度，取相似度最高的*n*个学者作为排名结果。



图 13 GARec算法整体框架

### 预训练论文向量

为了解决4.1节所述的“无穷用户集合问题”，需要一种能够将任意领域关键词编码为向量的方法，语言模型刚好能够解决这一问题。本文通过论文标题和关键词（领域）的对比学习对预训练的SciBERT模型[27]进行微调(fine-tune)，从而获得预训练的论文向量，如图 14所示。



图 14 通过对比学习微调SciBERT模型

输入*M*篇论文的标题和关键词，使用预训练的SciBERT模型分别将其编码为向量：



(.1)

其中和分别为第*i*篇论文的标题和关键词编码后的向量。之后计算对比损失并更新模型参数：



(.2)

其中*τ*为对比损失的温度参数。

注意到对比损失本质上就是交叉熵损失，相当于“样本”*ti*与“标签”*kj*匹配的概率，之后进行softmax归一化。因此公式(4.2)中第一个公式表达的含义为：将论文*i*的标题匹配到其关键词的交叉熵损失。第二个公式同理，只是反过来将关键词匹配到标题。通过这样的标题-关键词匹配任务，SciBERT模型能够学习到论文标题与关键词（领域）之间的对应关系，从而微调后的模型编码的标题向量包含了论文的语义信息，可用于后续的论文召回。

### 构造学者排名

本文利用有引用数的论文数据，分领域按引用数构造学者排名，将其作为真实排名用于训练GNN模型。

为了计算学者在某个领域的引用数，一种简单的方法是将学者在该领域所有论文的引用数求和，但这种方法假设一篇论文的所有作者有相同的贡献，忽略了作者顺序。为了更加合理地计算学者引用数，本文参考了AMiner发布的《AI 2000人工智能全球最具影响力学者榜单》[[8]](#footnote-9)中学者引用数的计算公式：假设一篇论文有*n*个作者，令第*m*作者的权重为1/*m*，最后一个视为通讯作者，权重为1/2，将该权重归一化之后对论文引用数加权求和。具体地，学者*i*在领域*k*的引用数使用下式计算：



(.3)

其中*cj*表示论文*j*的引用数，*Pi,k*为学者*i*在领域*k*的论文集合，*ri,j*表示学者*i*在论文*j*中的作者顺序。公式(4.3)对论文引用数取对数，避免高被引论文占绝对优势的问题。

根据以上公式，对领域*k*的论文关联的学者按引用数倒序排序并取top-n得到该领域的学者排名，将其作为真实排名(ground truth)用于后续的精排步骤。

### 论文召回

论文召回步骤的目的是筛选出与查询词相关的候选论文集合，过滤掉无用信息，缩小查询范围，如图 13 (a)所示。

使用4.2.1节微调后的SciBERT模型可以预先计算出所有论文的归一化标题向量。对于输入的任意领域关键词*k*，使用该模型将其编码为查询向量*q*，并与论文标题向量计算相似度（内积）：



(.4)

从而得到查询词与论文的相似度得分，取得分最高的*m*篇论文作为召回结果。

### 精排

精排步骤的目的是对于召回的论文关联的学者进行排名，如图 13 (b)所示。

对于领域*k*，从4.2.2节构造的真实学者排名中采样三元组，包括简单样本、困难样本和真-假样本三类：



(.5)

其中三元组表示学者*ip*在领域*k*的排名在学者*in*之前，*Ak*表示对于领域*k*召回的论文集合关联的学者集合。简单样本和困难样本的区别是*ip*和*in*的排名差距不同，分别用于学习差距较大和较小的学者之间的相对顺序。真-假样本是指*ip*属于领域*k*的真实排名，而*in*不属于，用于学习一个学者是否进入最终的top-n候选学者集合。

在训练过程中，对于每个领域*k*，使用4.2.3节所述的方法召回论文，得到论文集合。之后构造学者-论文二分子图，将其输入RHCO模型，得到网络结构编码器输出的学者顶点嵌入，之后使用三元组损失来更新模型参数：



(.6)

其中*sk*是根据领域*k*的名称使用微调后的SciBERT模型编码的向量，*d*表示向量的欧式距离，*α*为三元组损失的间隔参数。算法 1展示了模型训练过程。

|  |
| --- |
| **算法 1** GARec算法训练过程 |
| **输入：**异构图*G*，领域集合*K*，真实学者排名，论文召回结果，领域向量*S*  **输出：**学者嵌入*Zsc* |
| **function** GARec\_train(*G*, *K*, , , *S*)  初始化RHCO模型*f*  **for** epoch = 1 **to** *n*  **for** *k* **in** *K*  使用公式(4.5)采样三元组*Tk*  构造子图  *Zsc* = *f*(*Gk*, *Ak*)  使用公式(4.6)计算损失  反向传播、更新参数  *Zsc* = *f*(*G*, *A*)  **return** *Zsc* |

### 推断

精排步骤训练完成后，可以得到所有学者的顶点嵌入。对于输入的任意领域关键词，按上述方法召回论文并构造子图，使用查询向量*q*与学者顶点嵌入计算相似度（内积）：



(.7)

从而得到查询词与学者的相似度得分，取相似度最高的*n*个学者作为返回的排名结果。推断过程如算法 2所示。

|  |
| --- |
| **算法 2** GARec算法推断过程 |
| **输入：**微调后的SciBERT模型*f*SciBERT，论文标题向量*St*，学者嵌入*Zsc*，查询词query  **输出：**领域学者排名 |
| function GARec\_infer(*f*SciBERT, *St*, *Zsc*, query)  *q* = *f*SciBERT(query)  = recall(*St*, *q*)  *A* = 关联的学者集合  = rank(*Zsc*[*A*], *q*)  **return**  **function** recall(*St*, *q*)  使用公式(4.4)计算*Yp*  = *Yp*的top-*m*索引  **return**  **function** rank(*Zsc*, *q*)  使用公式(4.7)计算*Ya*  = *Ya*的top-*n*索引  **return** |

## 实验设计与分析

### 数据集

3.3.1节所述的ogbn-mag数据集在网络结构和数据规模上都符合领域学者排名任务的要求，但该数据集提供的是预处理过的图，只有顶点id而没有论文标题、学者姓名、领域名称等可视化系统所必需的原始数据。因此本文使用来自Open Academic Graph[[9]](#footnote-10)的微软学术原始数据，抽取了近10年的计算机领域论文子集，构造了一个异构学术网络oag-cs，网络结构如图 15所示。



图 15 oag-cs数据集网络结构

oag-cs数据集包含约420万顶点和3600万条边，规模大约是ogbn-mag数据集的2倍，具体数据如表 3所示。

表 3 oag-cs数据集

|  |  |  |
| --- | --- | --- |
| 数据集 | 顶点 | 边 |
| oag-cs | 学者(A): 2248205 | A-P: 6349317 |
| 论文(P): 1852225 | P-V: 1852225 |
| 期刊(V): 11177 | P-F: 17250107 |
| 机构(I): 13747 | P-P: 9194781 |
| 领域(F): 120992 | A-I: 1726212 |

本文从oag-cs数据集中筛选了论文数大于等于5000的306个领域，并使用4.2.2节所述的方法为每个领域构造了top 100学者排名，用于在精排步骤中训练RHCO模型。表 4展示了部分示例数据。

表 4 学者排名示例数据

|  |  |
| --- | --- |
| 领域 | 学者排名 |
| artificial intelligence | Christian Szegedy, Vincent Vanhoucke, Feiping Nie, ... |
| machine learning | Ilya Sutskever, Diederik P. Kingma, Geoffrey E. Hinton, ... |
| computer vision | Kaiming He, Ross Girshick, Jian Sun, ... |
| natural language processing | Christopher D. Manning, Jeffrey Pennington, Richard Socher, ... |
| robotics | Radu Bogdan Rusu, Jürgen Sturm, Emanuel Todorov, ... |
| ... | ... |

### 对比模型

GARec算法的对比模型包括：

**SciBERT**[27] 使用4.2.1节所述的方法微调SciBERT模型并计算论文标题和领域的向量，使用这两个向量的余弦相似度计算论文与领域的相似度得分，使用论文的相似度得分之和作为学者与领域的相似度得分。与GARec算法相比，这种方法在计算学者得分时仅利用了论文的语义信息，而忽略了异构图的结构信息。

**KGCN**[38] 该模型利用外部知识图谱辅助推荐算法，使用GAT聚集知识图谱中实体的邻居信息，从而获得用户相关的物品表示。在本文的研究问题中，学者排名数据相当于“用户-物品”历史交互信息，异构学术网络可用作知识图谱。与GARec算法相比，KGCN模型中使用的用户、实体和关系的表示是可学习参数，因此没有实际含义。另外，KGCN将知识图谱视为同构图，仅区分物品实体与其他实体之间的关系类型，忽略了异构图本身的顶点类型信息。

### 实现细节

对于GARec算法，RHCO模型的参数与第三章相同，SciBERT模型的输出维数为128，温度参数*τ* = 0.07，每个领域的真实排名中的学者数量，训练过程中每个领域采样的三元组数量|*Tk*| = 200，简单样本间隔*Measy* = 20，困难样本间隔*Mhard* = 5，每个领域召回的论文数量，三元组损失的间隔参数*α* = 0.02。训练过程迭代100次(epoch)，使用Adam优化器，学习率为0.001。

对于KGCN模型使用原论文中的参数设置，实体表示维数为16，模型层数为1，每个实体采样的邻居数量为8。训练过程迭代50次，批大小为256，使用Adam优化器，学习率为0.0005，正则化系数为0.0001。

### 评价指标

本文使用推荐系统中常用的两个评价指标Recall@k和nDCG@k作为领域学者排名的评价指标。召回率Recall@k表示所有相关结果在前*k*个推荐结果中出现的比例：



(.8)

其中TP@k表示前*k*个结果中相关的个数，FN@k表示不在前*k*个结果中的相关结果个数，TP@k + FN@k = P，即所有相关结果的个数。由于召回率是针对二分类问题定义的，因此本文将“相关”定义为学者在某个领域的真实top-n排名中出现，从而相关结果个数等于真实排名中的学者数量，即。

归一化折损累积增益(normalized discounted cumulative gain, nDCG)是一种常用的推荐结果评价指标。首先介绍折损累积增益(discounted cumulative gain, DCG)，其核心思想是高相关性的结果应当出现在更靠前的位置，引入了一个基于对数的惩罚函数来降低每个位置的相关性得分，定义如下：



(.9)

其中*ri*是第*i*个结果的相关性得分。由于不同查询返回的结果数量可能不同，为了允许跨查询比较，nDCG对DCG进行归一化：



(.10)

其中IDCG@k表示前*k*个推荐结果按最优方式排序的DCG得分。

nDCG的计算需要用到相关性得分的定义。本文将每个领域的真实排名中的学者数量设置为100。将学者按排名划分等级，每10名为一个等级，第1~10名为10分，第11~20名为9分，以此类推，如果不属于top-n排名则相关性得分为0。具体地，学者*i*与领域*k*的相关性得分按下式计算：



(.11)

使用公式(4.8)和(4.10)分别计算一个领域的召回率和nDCG得分，整体得分是所有领域得分的平均值。

### 搜索论文

使用公式(4.4)得到的论文召回结果与查询词具有较好的语义相关性，可用于实现学术推荐系统中的论文搜索功能。

表 5展示了“graph neural network”和“scholar disambiguation”两个查询词的top 10论文搜索结果。从表 5 (a)可以看出，“graph neural network”的搜索结果均包含这一关键词，这表明对于能够与论文标题通过关键词匹配的查询词，模型能够准确地搜索到相关论文。但搜索结果并不只是简单的关键词匹配。从表 5 (b)可以看出，“scholar disambiguation”的搜索结果并不包含这一关键词，而是包含“author name disambiguation”，这与查询词的含义是相同的。这一结果表明模型能够学习到这两个关键词之间的语义相关性，因此即使论文标题中不存在关键词“scholar disambiguation”，模型也能够搜索到相关论文。

表 5 论文搜索结果示例

(a) graph neural network

|  |  |
| --- | --- |
| 相似度 | 标题 |
| 0.9629 | Aggregation **Graph Neural Networks** |
| 0.9579 | Neural Graph Learning: Training **Neural Networks** Using **Graphs** |
| 0.9556 | Heterogeneous **Graph Neural Network** |
| 0.9552 | Neural Graph Machines: Learning **Neural Networks** Using **Graphs** |
| 0.9490 | On the choice of **graph neural network** architectures |
| 0.9474 | Measuring and Improving the Use of Graph Information in **Graph Neural Networks** |
| 0.9362 | Challenging the generalization capabilities of **Graph Neural Networks** for network modeling |
| 0.9295 | Strategies for Pre-training **Graph Neural Networks** |
| 0.9142 | Supervised **Neural Network** Models for Processing **Graphs** |
| 0.9112 | Geometrically Principled Connections in **Graph Neural Networks** |

(b) scholar disambiguation

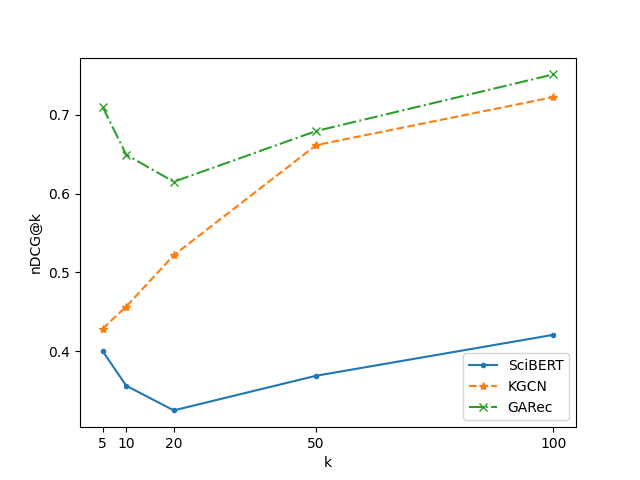
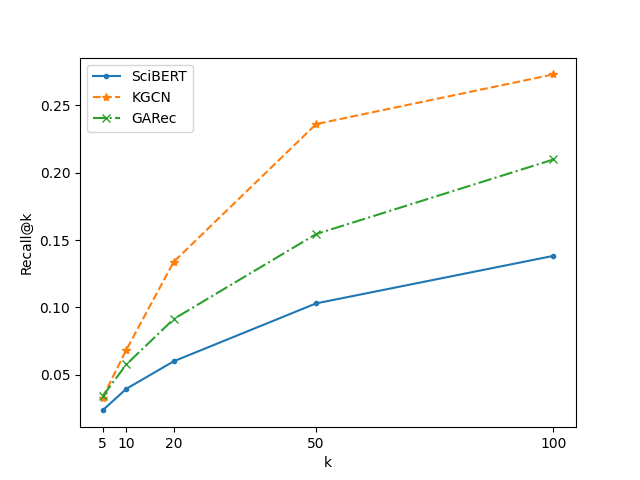
|  |  |
| --- | --- |
| 相似度 | 标题 |
| 0.9690 | Scholar search-oriented **author disambiguation** |
| 0.9040 | **Author name disambiguation** in scientific collaboration and mobility cases |
| 0.8901 | Exploring **author name disambiguation** on PubMed-scale |
| 0.8852 | **Author Name Disambiguation** in Heterogeneous Academic Networks |
| 0.8797 | KDD Cup 2013: **author disambiguation** |
| 0.8796 | A survey of **author name disambiguation** techniques: 2010–2016 |
| 0.8721 | Who is Who: **Name Disambiguation** in Large-Scale Scientific Literature |
| 0.8660 | Use of ResearchGate and Google CSE for **author name disambiguation** |
| 0.8643 | Automatic Methods for **Disambiguating Author Names** in Bibliographic Data Repositories |
| 0.8641 | A brief survey of automatic methods for **author name disambiguation** |

### 学者排名

为了评价模型在学者排名任务上的性能，将模型输出的学者得分与真实学者排名计算nDCG@k，对得分取top-n得到预测学者列表与真实学者排名计算Recall@k，其中k的取值范围为[5, 10, 20, 50, 100]。各模型的评价指标如表 6和图 16所示。

表 6 学者排名实验结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 评价指标 | k | SciBERT | KGCN | GARec |
| nDCG@k | 5 | 0.4003 | 0.4286 | **0.7105** |
| 10 | 0.3562 | 0.4568 | **0.6495** |
| 20 | 0.3249 | 0.5219 | **0.6152** |
| 50 | 0.3689 | 0.6617 | **0.6796** |
| 100 | 0.4209 | 0.7227 | **0.7515** |
| Recall@k | 5 | 0.0235 | 0.0330 | **0.0339** |
| 10 | 0.0395 | **0.0684** | 0.0576 |
| 20 | 0.0600 | **0.1339** | 0.0914 |
| 50 | 0.1030 | **0.2362** | 0.1545 |
| 100 | 0.1383 | **0.2731** | 0.2099 |

(a) nDCG@k (b) Recall@k

图 16 学者排名实验结果

从结果中可以看出，GARec和KGCN在两个指标上均远高于SciBERT，这表明在推荐算法中利用图结构信息（知识图谱）是有帮助的。另外，GARec在nDCG上高于KGCN，而在召回率上低于KGCN，这表明KGCN能够更加准确地召回相关学者，而GARec对于召回的学者能够更好地排名。在学者排名任务中，后者往往更加重要。

图 16 (a)显示，随着k的增大，GARec的nDCG指标先下降后上升，而KGCN是一直上升。这表明对于排名靠前的学者，GARec能够更好地排序。当k = 5或10时GARec的nDCG指标远高于KGCN，即仅考虑排名结果的前5名或前10名时，GARec生成的结果优于KGCN，而这也是用户更加关注的。

### 参数敏感性分析

三元组损失是一种被广泛应用的度量学习损失，如公式(4.6)所示。输入的三元组包括一个固定样本a(anchor)、正样本p和负样本n。其核心思想是p与a之间的距离应当尽可能近，n与a之间的距离应当尽可能远。这两个距离之间有一个最小间隔*α*，是一个重要的参数，当时就会产生损失。因此当*α*越小时，损失越容易趋近于0，训练结果不能很好地区分正负样本；当*α*越大时，损失越难以趋近于0，甚至可能导致模型不收敛。因此设置一个合适的*α*值至关重要。

本文分析了*α*对GARec算法性能的影响，*α*的取值范围为[0.001, 0.002, 0.005, 0.01, 0.02, 0.05, 0.1, 0.2, 0.5, 1]，观察的评价指标为nDCG@100和Recall@100，结果如图 17所示。

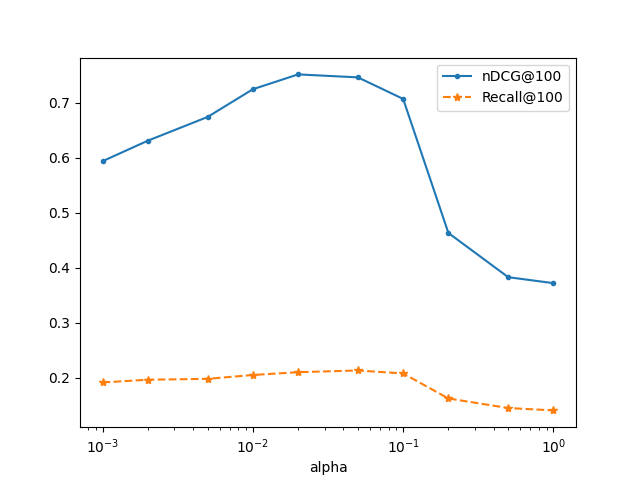


图 17 GARec算法参数敏感性分析实验结果

从结果中可以看出，*α*在0.02~0.05范围内取得最好结果。*α*过小时，模型不能很好地区分不同学者与领域的相似度，返回的结果接近于随机排名；*α*过大时，对学者区分度的要求过高，由于学者排名的训练数据有限，模型难以拟合出较好的结果。由于在学者排名任务中nDCG指标更重要，因此本文将*α*设置为0.02。

### 案例分析

本节在两个不同场景下对GARec算法的学者排名结果进行案例分析，从而验证该算法的有效性。

（1）大类领域

对于计算机领域下的大类领域，有一些公认的影响力较高的学者，排名的歧义较小。对于这些领域，推荐算法应当能够较准确地识别出这些学者。

AMiner发布的AI 2000学者榜单包含20个人工智能子领域的学者排名，具有较高的可信度。因此本文将计算机视觉领域GARec算法返回的top 10学者排名与AI 2000的排名进行对比，如表 7所示。可以看出，AI 2000的前7名均在GARec返回的前10名中出现，且相对顺序具有较好的一致性。这表明对于大类领域，GARec算法能够准确地识别出具有较高影响力的学者。

表 7 “计算机视觉”学者排名案例分析

|  |  |  |  |
| --- | --- | --- | --- |
| 排名 | 姓名 | 引用数 | AI 2000排名 |
| 1 | Kaiming He | 92894 | 1 |
| 2 | Karen Simonyan | 36891 | - |
| 3 | Jian Sun | 85956 | 3 |
| 4 | Andrew Zisserman | 59859 | 33 |
| 5 | Ross Girshick | 82619 | 2 |
| 6 | Xiangyu Zhang | 53905 | 5 |
| 7 | Trevor Darrell | 58876 | 7 |
| 8 | Shaoqing Ren | 65060 | 4 |
| 9 | Christian Szegedy | 40761 | 6 |
| 10 | Li Fei-Fei | 37977 | 22 |

（2）数据集中不存在的领域

GARec算法与传统的推荐算法最大的不同在于，传统推荐算法只能对数据集中存在的“用户”（在这里相当于领域）进行推荐，而GARec算法基于语义相似度进行论文召回和学者排名，能够针对用户输入的任意领域关键词进行学者排名，即解决了4.1节所述的“无穷用户集合问题”。

表 8展示了GARec算法针对“基于图神经网络的推荐系统”这一关键词返回的top 5学者排名，以及每个学者在GNN和推荐系统领域各自的代表论文。该查询词是“图神经网络”和“推荐系统”的交叉领域，并不是数据集中存在的领域，其排名结果应当包含在这两个领域都有所涉及的学者。从结果中可以看出，前3名的学者在推荐系统领域的代表论文都是“Graph Convolutional Neural Networks for Web-Scale Recommender Systems”，该论文恰好属于“基于图神经网络的推荐系统”这一交叉领域。另外，前2名的学者各有一篇经典的GNN论文（GraphSAGE和node2vec）。第5名学者的代表论文“KGAT: Knowledge Graph Attention Network for Recommendation”也属于“基于图神经网络的推荐系统”这一领域，结果表明GARec算法返回的学者的研究领域与查询词具有较高的匹配度。因此GARec算法对于数据集中不存在的领域也能生成较好的排名结果，而使用传统的推荐算法（例如KGCN）则无法实现这样的推荐。

表 8 “基于图神经网络的推荐系统”学者排名案例分析

|  |  |  |  |
| --- | --- | --- | --- |
| 排名 | 姓名 | GNN代表论文 | 推荐系统代表论文 |
| 1 | Rex Ying | Inductive Representation Learning on Large Graphs | Graph Convolutional Neural Networks for Web-Scale Recommender Systems |
| 2 | Jure Leskovec | node2vec: Scalable Feature Learning for Networks | Graph Convolutional Neural Networks for Web-Scale Recommender Systems |
| 3 | Ruining He | - | Graph Convolutional Neural Networks for Web-Scale Recommender Systems |
| 4 | Muhan Zhang | Link Prediction Based on Graph Neural Networks | Active learning for ranking with sample density |
| 5 | Xiang Wang | - | KGAT: Knowledge Graph Attention Network for Recommendation |

## 小结

本章介绍了基于图神经网络的学术推荐算法GARec。该算法通过论文标题和关键词的对比学习对预训练的SciBERT模型进行微调，从而获得论文标题向量；利用论文标题向量计算查询词与论文的相似度得分，实现论文召回；利用RHCO模型学习到的学者顶点嵌入计算查询词与学者的相似度得分，实现学者排名。该算法解决了学者排名问题中存在的“无穷用户集合问题”。定量实验和案例分析证明了该算法的有效性。

# 学术推荐系统设计与实现

## 系统概述

在如今的大数据时代，互联网上存在海量的科研文献数据。在过去的几个世纪中，科研文献的数量呈指数级增长，每个月都有几百万篇新的文献发表。对于学术研究者来说，快速并准确地找到与自己的研究方向相关的论文和学者能够节省大量的时间。Google学术、微软学术等学术搜索引擎提供了搜索论文、领域学者排名等功能。然而现有的学者排名功能大多是基于关键词匹配和引用数统计分析实现的，对于数据库中不存在的领域关键词或者没有引用数的论文数据难以进行准确的排名和推荐。

本文以4.3.1节所述的来自Open Academic Graph的近10年计算机领域论文数据集oag-cs为基础，使用GARec算法实现领域学者排名功能，并实现了学术推荐系统，对本文的研究成果进行可视化展示。该系统是一个基于浏览器/服务器(B/S)架构的Web应用，开发环境为PyCharm 2021.2，开发语言使用Python 3.7和HTML，Web框架使用Django 3.2，前端页面样式使用Bootstrap 4.3，数据库使用MySQL 8.0。



图 18 MTV模式操作流程图

Django[[10]](#footnote-11)是一个使用Python编写的开源Web应用框架。Django提供了开发Web应用所需的大部分功能，因此开发人员可以专注于应用本身的业务逻辑而无需编写大量重复的样板代码。与传统的MVC (Model-View-Controller)模式不同，Django的设计遵循MTV (Model-Template-View)模式。模型(model)即数据存取层，负责业务对象与数据库的对象关系映射(object relational mapping, ORM)以及数据库查询操作；模板(template)即展示给用户的HTML页面；视图(view)负责处理HTTP请求、执行业务逻辑、渲染模板以及返回HTTP响应。MTV模式的操作流程图如图 18所示。

## 总体设计

学术推荐系统采用分层架构，自底向上分为数据持久化层、业务逻辑层和展示层，整体架构图如图 19所示。



图 19 学术推荐系统架构图

各层的主要功能如下：

（1）数据持久化层：负责提供系统需要的数据。使用MySQL存储oag-cs数据集中的学者、论文、领域、期刊、机构原始数据及其关联关系。另外，以文件形式存储4.2.3节中预先计算好的论文标题向量*St*和4.2.4节中训练得到的学者顶点嵌入*Zsc*，供业务逻辑层使用。



图 20 系统E-R图

（2）业务逻辑层：负责实现系统的核心业务逻辑，由用户管理、论文召回和学者排名三个核心模块组成。其中用户管理模块负责用户注册、登录、注销功能，由Django内置的认证功能实现，因此不再详细介绍；论文召回模块使用4.2.3节中GARec算法的论文召回步骤，根据用户输入的领域关键词召回相关论文并返回论文列表；学者排名模块使用4.2.4节中GARec算法的精排步骤，根据用户输入的领域关键词召回论文，之后对于召回的论文关联的学者计算相似度得分并排名，返回学者列表。

（3）展示层：负责提供用户与系统交互的前端界面。使用Django的模板系统编写HTML页面，主要包括用户注册、用户登录、搜索论文、搜索学者、查看论文详情和查看学者详情六个页面。其中用户注册/登录调用用户管理模块，搜索论文和搜索学者分别调用论文召回和学者排名模块。查看论文详情和查看学者详情分别用于展示论文和学者的详细信息，可使用Django提供的详情视图实现。

## 详细设计

### 数据库设计

oag-cs数据集是图结构的数据，顶点之间通过边相连。为了能够将其导入关系型数据库，需要将图结构数据转换为关系型数据。具体地，学者、论文、领域、期刊、机构这五种类型的顶点分别对应一张数据库表；P-V边和A-I边是一对多关系，因此转换为一对多外键，可分别在论文表和学者表中用一个外键字段表示；A-P边、P-F边和P-P边是多对多关系，因此转换为多对多外键，每种边分别对应一张数据库表，其中包含两端顶点id的外键字段。整体的E-R图如图 20所示。

数据库创建完成后，编写Python脚本读取JSON格式的原始数据并导入MySQL数据库。

### 用例图

基于5.2节中的分析，系统包含注册、登录、搜索论文、搜索学者、查看论文详情和查看学者详情六个用例，系统用例图如图 21所示。其中搜索学者用例包含对论文召回功能的调用，因此包含搜索论文用例。



图 21 系统用例图

### 论文召回模块

该模块首先使用4.2.1节微调后的SciBERT模型将用户输入的查询词编码为向量，之后使用公式(4.4)计算查询词与论文的相似度得分，返回得分最高的20篇论文，流程图如图 22 (a)所示。

(a) 论文召回模块 (b) 学者排名模块

图 22 系统流程图

### 学者排名模块

学者排名模块首先调用论文召回模块根据用户输入的查询词召回论文，之后从数据库查询出这些论文关联的学者，最后使用公式(4.7)计算查询词与学者的相似度得分，返回得分最高的100名学者，流程图如图 22 (b)所示。

## 效果展示

本节将对学术推荐系统的各功能模块进行效果展示。

用户注册和登录界面分别如图 23和图 24所示。由于系统限制了其他功能登录后才能访问，因此未登录的用户访问其他页面都将被重定向到登录页面。用户登录后进入系统首页，如图 25所示。

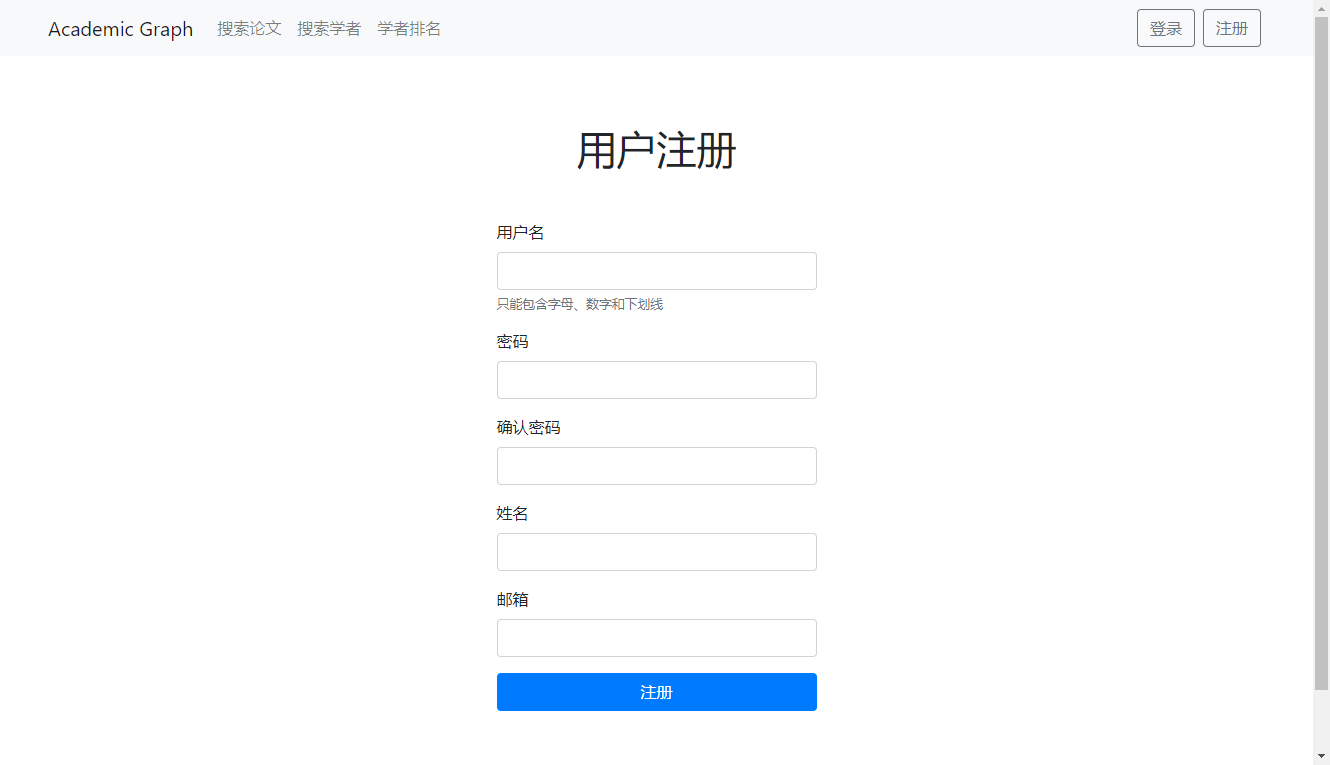


图 23 用户注册页面



图 24 用户登录页面

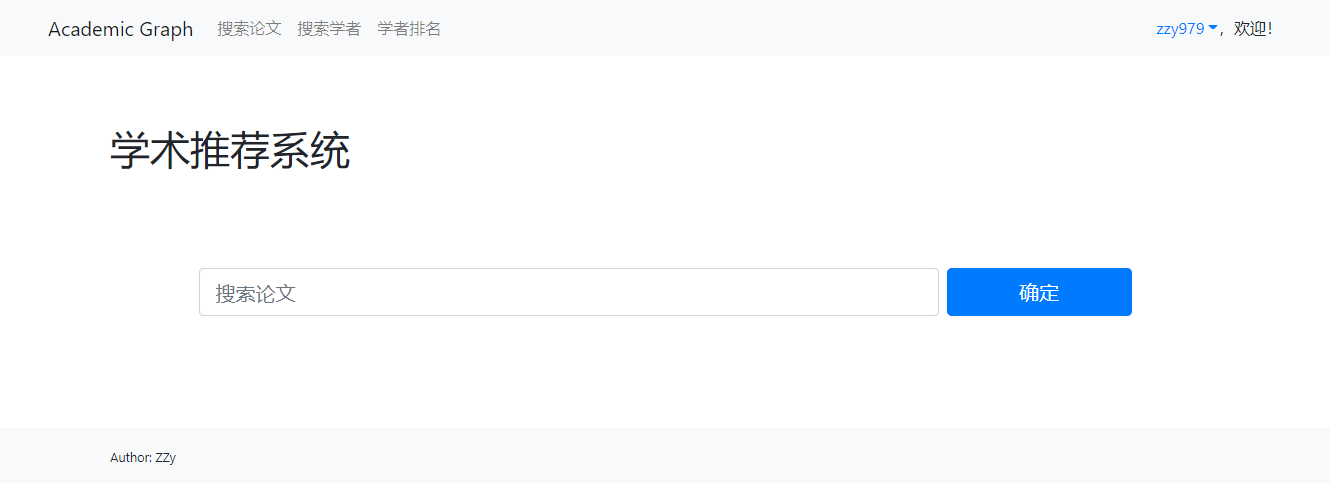
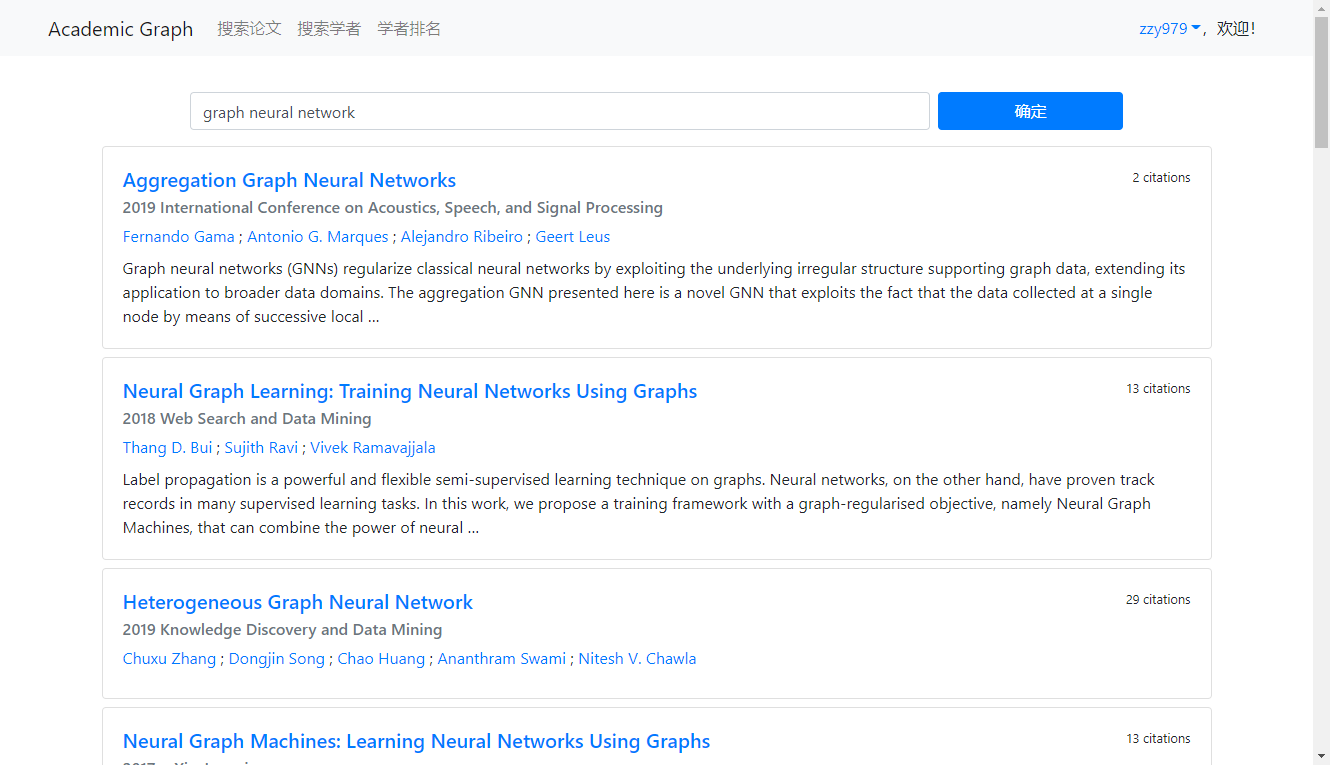
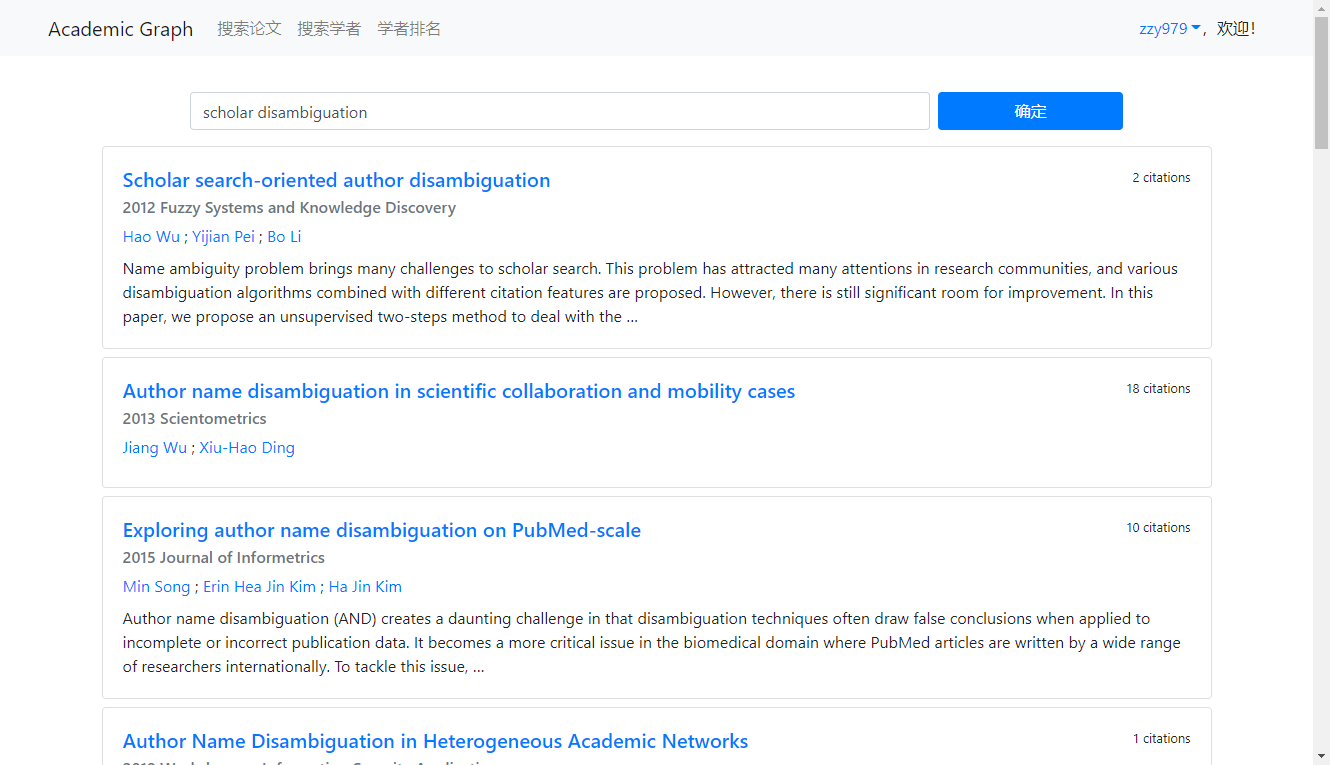


图 25 系统首页

图 26 (a)是关键词“graph neural network”的论文搜索结果，从结果中可以看出，召回的论文与查询词有较高的相关性。图 26 (b)是关键词“scholar disambiguation”的论文搜索结果，从结果中可以看出，召回的论文并不都包含关键词“scholar disambiguation”，而是包含“author name disambiguation”，这与查询词在语义上高度相关。



(a) graph neural network



(b) scholar disambiguation

图 26 论文搜索结果

图 27是论文详情页面，其中展示了论文的标题、期刊、作者、引用数、摘要、领域等基本信息。

图 28是关键词“computer vision”的学者搜索结果，图 29是排名第一的学者Kaiming He的详情页面，其中展示了作者的机构、引用数和部分论文（按引用数降序排序）。从论文中可以看出，该学者在CVPR, NeurIPS, ICCV等国际会议上发表过多篇高被引论文，因此该学者作为计算机视觉领域推荐结果中排名第一具有较高的置信度。

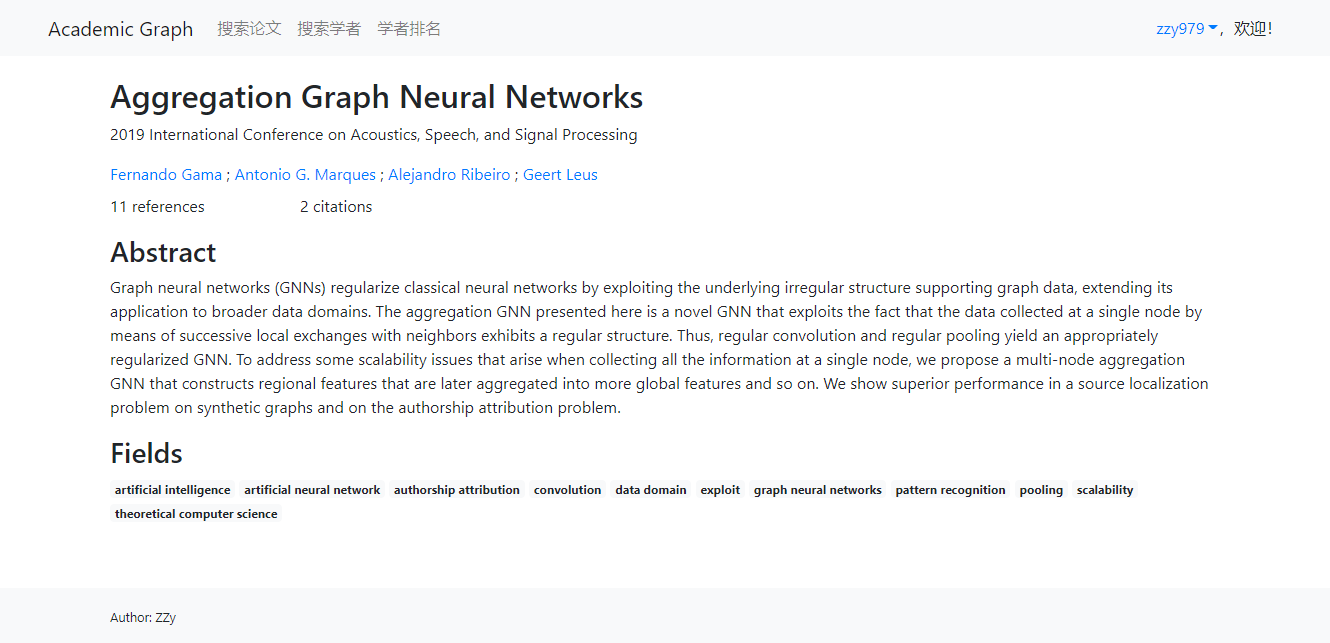


图 27 论文详情页面

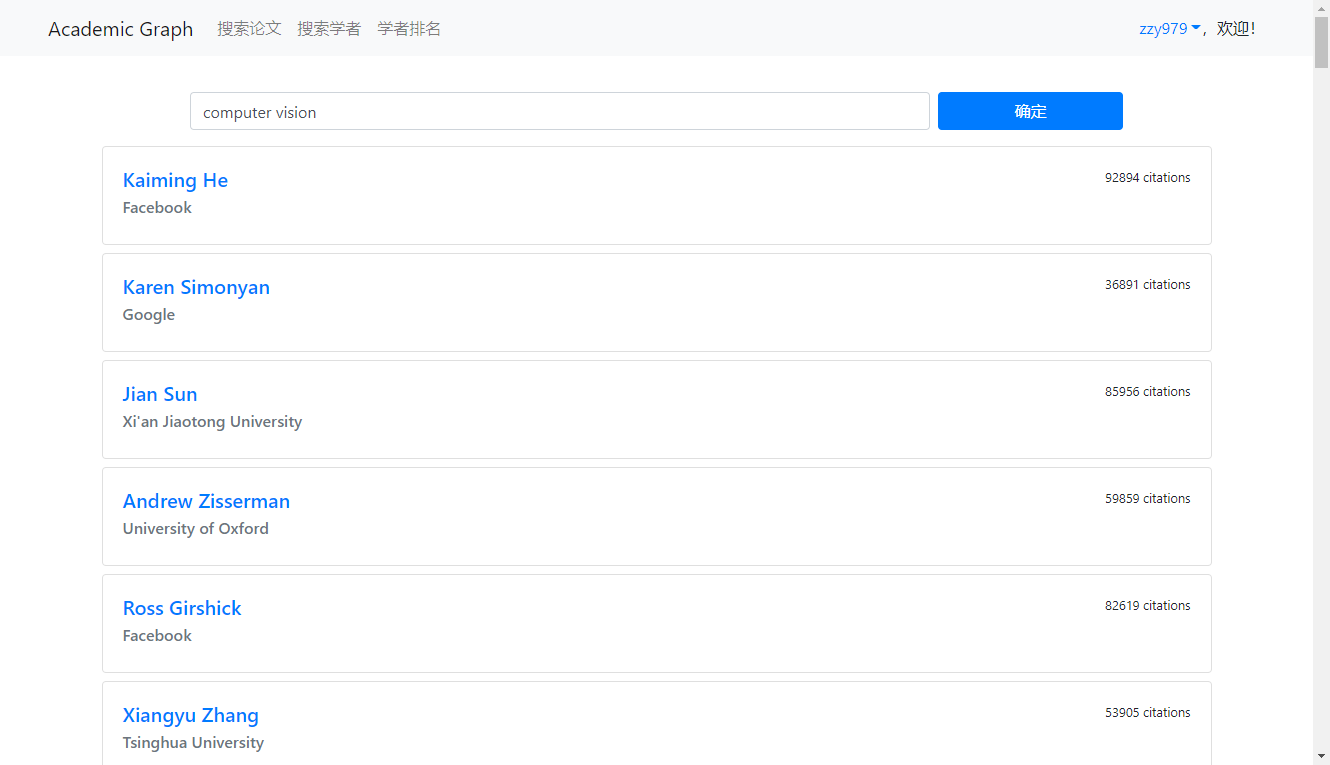


图 28 学者搜索结果

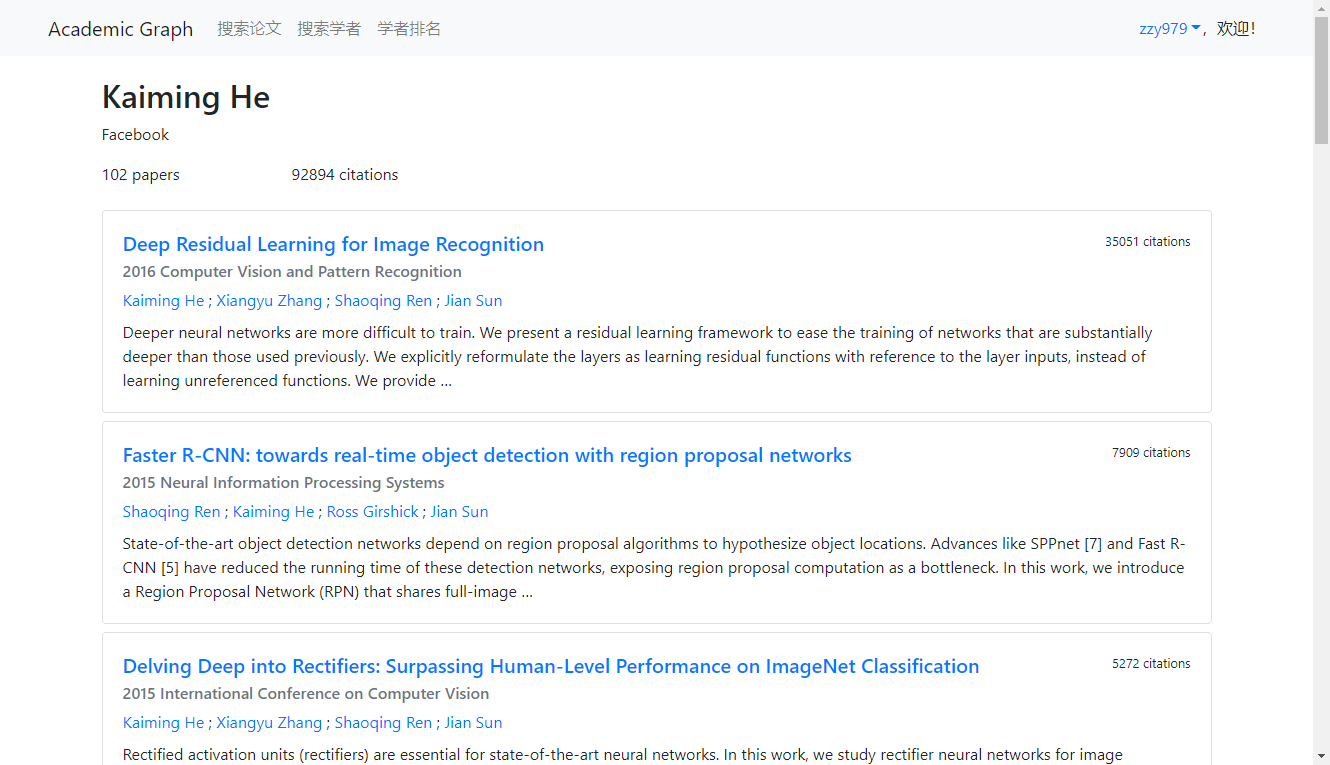


图 29 学者详情页面

## 小结

本章介绍了用于搜索论文和学者排名的学术推荐系统，包括系统概述、总体设计以及各功能模块的详细设计。本章基于计算机领域的论文构造的学术网络数据集，应用本文提出的GARec算法实现了论文召回和领域学者排名两个核心功能，最后通过系统演示展示了推荐算法的效果。

# 总结与展望

## 总结

图神经网络是一种强大的基于深度学习的图表示学习方法。本文针对基于图神经网络的异构图表示学习和推荐算法展开研究，针对现有的异构图表示学习方法可扩展性差、输入特征简单等问题，提出了基于对比学习的关系感知异构图神经网络模型RHCO；针对学术推荐任务中的领域学者排名问题中存在的“无穷用户集合问题”，提出了基于图神经网络的学术推荐算法GARec；以GARec算法为基础，设计并实现了学术推荐系统。

本文的主要工作如下：

（1）针对异构图表示学习任务，在HeCo模型的基础上进行了改进，提出了基于对比学习的关系感知异构图神经网络模型RHCO。该模型在顶点嵌入中融入了关系信息，使得输出的嵌入能够包含关系信息；使用预训练的HGT计算的注意力权重选择正样本，避免了显式地构造基于元路径的邻居图，使得模型能够扩展到大规模数据集；通过增加标签平滑处理提升了模型在顶点分类任务上的性能。

（2）针对领域学者排名问题中存在的“无穷用户集合问题”，提出了基于图神经网络的学术推荐算法GARec。该算法通过论文标题和关键词的对比学习对预训练的SciBERT模型进行微调，从而获得论文标题向量；利用论文标题向量计算查询词与论文的相似度得分，实现论文召回；利用RHCO模型学习到的学者顶点嵌入计算查询词与学者的相似度得分，实现学者排名。定量实验和案例分析证明了该算法的有效性。

（3）基于Django框架构建了学术推荐系统，用于对论文召回和学者排名功能进行可视化展示。该系统旨在帮助学术研究者快速并准确地找到与自己的研究方向相关的论文和学者，从而能够节省大量的时间。该系统应用GARec算法实现了论文召回和领域学者排名，以及查看论文详情和学者详情等功能。

## 展望

本文在基于图神经网络的异构图表示学习和推荐算法领域取得了一定的成果。但是在研究过程中发现仍然存在一些问题，值得进一步研究：

（1）RHCO模型构造正样本图的步骤比较复杂。需要预先训练一个HGT模型，之后再对数据集进行一次遍历，利用HGT计算出的注意力权重选择每个顶点的正样本。之后可以研究如何设计一种更加简单，同时能够用于大规模数据集的正样本选择策略。

（2）推荐算法的精排步骤依赖于人工标注。为了训练GNN模型需要学者的真实排名数据。虽然可以通过引用数求和构造每个领域的学者排名，但这样构造的排名不一定完全准确，仍然需要人工判断并标注一些准确的排名数据。另外，在构造真实排名时也可以考虑论文的年份，不同年份的引用数权重不同。例如发表2年、引用数为1000的论文比发表10年、引用数为3000的论文具有更高的热度和影响力。

（3）推荐系统暂不支持新增学者、论文或领域。目前的推荐系统使用的是“静态”图，即图中的顶点和边不会变化。然而真实的学术网络是在一直演变的，随着时间的推移会增加新的顶点和边。为了支持动态变化的数据，需要研究如何对新增的学者顶点进行增量式训练，从而获得新增学者的顶点嵌入。

# 参考文献

1. Scarselli F, Gori M, Tsoi A C, et al. The graph neural network model[J]. IEEE Transactions on Neural Networks, 2008, 20(1): 61-80.
2. Yang C, Xiao Y, Zhang Y, et al. Heterogeneous network representation learning: A unified framework with survey and benchmark[J]. IEEE Transactions on Knowledge and Data Engineering, 2020.
3. Shen Z, Ma H, Wang K. A web-scale system for scientific knowledge exploration[J]. arXiv preprint arXiv:1805.12216, 2018.
4. Gilmer J, Schoenholz S S, Riley P F, et al. Neural message passing for quantum chemistry[J]. arXiv preprint arXiv:1704.01212, 2017.
5. Perozzi B, Al-Rfou R, Skiena S. Deepwalk: Online learning of social representations[A]. Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining[C]. 2014: 701-710.
6. Grover A, Leskovec J. node2vec: Scalable feature learning for networks[A]. Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining[C]. 2016: 855-864.
7. Narayanan A, Chandramohan M, Venkatesan R, et al. graph2vec: Learning distributed representations of graphs[J]. arXiv preprint arXiv:1707.05005, 2017.
8. Dong Y, Chawla N V, Swami A. metapath2vec: Scalable representation learning for heterogeneous networks[A]. Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining[C]. 2017: 135-144.
9. Fu T, Lee W C, Lei Z. Hin2vec: Explore meta-paths in heterogeneous information networks for representation learning[A]. Proceedings of the 2017 ACM on Conference on Information and Knowledge Management[C]. 2017: 1797-1806.
10. Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[J]. arXiv preprint arXiv:1609.02907, 2016.
11. Schlichtkrull M, Kipf T N, Bloem P, et al. Modeling relational data with graph convolutional networks[A]. European Semantic Web Conference[C]. Springer, Cham, 2018: 593-607.
12. Hamilton W, Ying Z, Leskovec J. Inductive representation learning on large graphs[A]. Advances in neural information processing systems[C]. 2017: 1024-1034.
13. Veličković P, Cucurull G, Casanova A, et al. Graph attention networks[J]. arXiv preprint arXiv:1710.10903, 2017.
14. Zhang C, Song D, Huang C, et al. Heterogeneous graph neural network[A]. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining[C]. 2019: 793-803.
15. Wang X, Ji H, Shi C, et al. Heterogeneous graph attention network[A]. The World Wide Web Conference[C]. 2019: 2022-2032.
16. Fu X, Zhang J, Meng Z, et al. MAGNN: Metapath Aggregated Graph Neural Network for Heterogeneous Graph Embedding[A]. Proceedings of The Web Conference 2020[C]. 2020: 2331-2341.
17. Hu Z, Dong Y, Wang K, et al. Heterogeneous graph transformer[A]. Proceedings of The Web Conference 2020[C]. 2020: 2704-2710.
18. Yu L, Sun L, Du B, et al. Hybrid Micro/Macro Level Convolution for Heterogeneous Graph Learning[J]. arXiv preprint arXiv:2012.14722, 2020.
19. Yu L, Sun L, Du B, et al. Heterogeneous Graph Representation Learning with Relation Awareness[J]. arXiv preprint arXiv:2105.11122, 2021.
20. Huang Q, He H, Singh A, et al. Combining label propagation and simple models out-performs graph neural networks[J]. arXiv preprint arXiv:2010.13993, 2020.
21. Zhu X, Ghahramani Z. Learning from labeled and unlabeled data with label propagation[J]. 2002.
22. Wang X, Liu N, Han H, et al. Self-supervised Heterogeneous Graph Neural Network with Co-contrastive Learning[J]. arXiv preprint arXiv:2105.09111, 2021.
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. Mikolov T, Sutskever I, Chen K, et al. Distributed representations of words and phrases and their compositionality[A]. Advances in neural information processing systems[C]. 2013: 3111-3119.
25. Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30: 5998-6008.
26. Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.
27. Beltagy I, Lo K, Cohan A. Scibert: A pretrained language model for scientific text[J]. arXiv preprint arXiv:1903.10676, 2019.
28. Wu S, Sun F, Zhang W, et al. Graph neural networks in recommender systems: a survey[J]. arXiv preprint arXiv:2011.02260, 2020.
29. Adomavicius G, Tuzhilin A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions[J]. IEEE transactions on knowledge and data engineering, 2005, 17(6): 734-749.
30. Berg R, Kipf T N, Welling M. Graph convolutional matrix completion[J]. arXiv preprint arXiv:1706.02263, 2017.
31. Zhang J, Shi X, Zhao S, et al. Star-gcn: Stacked and reconstructed graph convolutional networks for recommender systems[J]. arXiv preprint arXiv:1905.13129, 2019.
32. Wang X, He X, Wang M, et al. Neural graph collaborative filtering[A]. Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval[C]. 2019: 165-174.
33. Ying R, He R, Chen K, et al. Graph convolutional neural networks for web-scale recommender systems[A]. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining[C]. 2018: 974-983.
34. Wu L, Sun P, Fu Y, et al. A neural influence diffusion model for social recommendation[A]. Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval[C]. 2019: 235-244.
35. Fan W, Ma Y, Li Q, et al. Graph neural networks for social recommendation[A]. The World Wide Web Conference[C]. 2019: 417-426.
36. Wu Q, Zhang H, Gao X, et al. Dual graph attention networks for deep latent representation of multifaceted social effects in recommender systems[A]. The World Wide Web Conference[C]. 2019: 2091-2102.
37. Wu L, Li J, Sun P, et al. Diffnet++: A neural influence and interest diffusion network for social recommendation[J]. IEEE Transactions on Knowledge and Data Engineering, 2020.
38. Wang H, Zhao M, Xie X, et al. Knowledge graph convolutional networks for recommender systems[A]. The world wide web conference[C]. 2019: 3307-3313.
39. Wang H, Zhang F, Zhang M, et al. Knowledge-aware graph neural networks with label smoothness regularization for recommender systems[A]. Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining[C]. 2019: 968-977.
40. Wang X, He X, Cao Y, et al. Kgat: Knowledge graph attention network for recommendation[A]. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining[C]. 2019: 950-958.
41. Wu S, Tang Y, Zhu Y, et al. Session-based recommendation with graph neural networks[A]. Proceedings of the AAAI Conference on Artificial Intelligence[C]. 2019, 33(01): 346-353.
42. Wu S, Zhang M, Jiang X, et al. Personalizing graph neural networks with attention mechanism for session-based recommendation[J]. arXiv preprint arXiv:1910.08887, 2019.
43. Qiu R, Li J, Huang Z, et al. Rethinking the item order in session-based recommendation with graph neural networks[A]. Proceedings of the 28th ACM International Conference on Information and Knowledge Management[C]. 2019: 579-588.
44. Ma C, Ma L, Zhang Y, et al. Memory augmented graph neural networks for sequential recommendation[A]. Proceedings of the AAAI Conference on Artificial Intelligence[C]. 2020, 34(04): 5045-5052.
45. Song W, Xiao Z, Wang Y, et al. Session-based social recommendation via dynamic graph attention networks[A]. Proceedings of the Twelfth ACM international conference on web search and data mining[C]. 2019: 555-563.
46. Li Q, Han Z, Wu X M. Deeper insights into graph convolutional networks for semi-supervised learning[A]. Thirty-Second AAAI conference on artificial intelligence[C]. 2018.
47. Rumelhart D E, Hinton G E, Williams R J. Learning representations by back-propagating errors[J]. Nature, 1986, 323(6088): 533-536.
48. Liu X, Zhang F, Hou Z, et al. Self-supervised learning: Generative or contrastive[J]. IEEE Transactions on Knowledge and Data Engineering, 2021.
49. Peng Z, Dong Y, Luo M, et al. Self-supervised graph representation learning via global context prediction[J]. arXiv preprint arXiv:2003.01604, 2020.
50. Wang P, Agarwal K, Ham C, et al. Self-Supervised Learning of Contextual Embeddings for Link Prediction in Heterogeneous Networks[A]. Proceedings of the Web Conference 2021[C]. 2021: 2946-2957.
51. 周连明. 异质图嵌入技术及其应用研究[D].电子科技大学,2020.
52. 潘林林. 基于异构图的学术论文推荐系统[D].南京大学,2015.
53. 秦川,祝恒书,庄福振,郭庆宇,张琦,张乐,王超,陈恩红,熊辉.基于知识图谱的推荐系统研究综述[J].中国科学:信息科学,2020,50(07):937-956.
54. 荣沛,苏凡军.基于知识图注意网络的个性化推荐算法[J/OL].计算机应用研究:1-5[2020-12-17].https://doi.org /10.19734/j.issn.1001-3695.2020.02.0015.
55. 饶子昀,张毅,刘俊涛,曹万华.应用知识图谱的推荐方法与系统[J/OL].自动化学报:1-17[2020-12-17].https://doi.org /10.16383/j.aas.c200128.

# 攻读硕士学位期间取得的学术成果

1. Xiaowei Zhao, Deqing Wang, Zhengyang Zhao, Wei Liu, Chenwei Lu, Fuzhen Zhuang. A neural topic model with word vectors and entity vectors for short texts[J]. Information Processing & Management, 2021, 58(2): 102455.
2. Ting Jiang, Deqing Wang, Leilei Sun, Huayi Yang, Zhengyang Zhao, Fuzhen Zhuang. LightXML: Transformer with Dynamic Negative Sampling for High-Performance Extreme Multi-label Text Classification[A]. Proceedings of the AAAI Conference on Artificial Intelligence[C]. 2021, 35(9): 7987-7994.

# 致谢

时光飞逝，两年半的研究生生活已经接近尾声。在这两年半的时间里，虽免不了困难和坎坷，也有很多收获。经过不断的学习和进步，我的专业技能和科研能力都有了很大提升，为人处世、价值观等方面也有了很大的变化。这一切都离不开导师、同学和家人对我的支持和帮助。在这里，我真诚地向他们表示感谢。

首先衷心地感谢我的导师王德庆老师。在我的研究生期间，王老师无论在学习、科研还是生活上，都给予了我非常大的帮助。在项目工作中，王老师提出了很多宝贵意见，使我的专业技能有了很大提升；在科研方面，王老师指导我确定研究方向、开展科研，使我学会了独立阅读论文和研究问题；在生活方面，王老师经常关心我们的生活，使我们能够专心投入学习和科研工作中。感谢庄福振老师在科研上对我们的督促和指导，经常与我们分享宝贵的科研经验，帮助我们少走弯路。

同时，还要感谢实验室的杜晨光学长和赵晓薇师姐，感谢你们在科研和项目工作中对我的帮助。感谢实验室的杨华易、蒋挺、吴嘉伟、周昊、郭浩隆，在我的论文研究过程中经常一起探讨问题，给我带来了很多启发，另外在我找工作期间也给予了我很多的帮助。感谢实验室的万梓明师弟、明雪华师妹、耿皓师弟、王宁境师妹，感谢你们在学习和生活中对我的帮助。

最后，感谢北航对我的培养，给我提供了一个成长的平台，感谢软件开发环境国家重点实验室提供的资源支持。感谢父母对我的养育之恩，感谢一直在背后关心和支持我的家人和朋友。这两年半的研究生生活充实而愉快，将是我人生中最难忘的时光。

1. <https://scholar.google.com/> [↑](#footnote-ref-2)
2. <https://academic.microsoft.com/home> [↑](#footnote-ref-3)
3. <http://kejso.com/> [↑](#footnote-ref-4)
4. <https://github.com/liun-online/HeCo/tree/main/data/dblp> [↑](#footnote-ref-5)
5. <https://ogb.stanford.edu/docs/nodeprop/#ogbn-mag> [↑](#footnote-ref-6)
6. <https://pytorch.org/> [↑](#footnote-ref-7)
7. <https://www.dgl.ai/> [↑](#footnote-ref-8)
8. <https://www.aminer.cn/ai2000> [↑](#footnote-ref-9)
9. <https://www.aminer.cn/oag-2-1> [↑](#footnote-ref-10)
10. <https://www.djangoproject.com/> [↑](#footnote-ref-11)