



# 河南财经政法大学

本科生毕业论文

## 基于图神经网络的商品推荐 冷启动研究

姓 名	宋世超
学 号	201934170452
专 业	电子商务
班 级	191704 班
学 院	电子商务与物流管理学院
指导教师	潘勇 教授
完成时间	2023 年 5 月 20 日

## 摘要

推荐系统发展至今，有些问题仍然困扰着学术界和工业界，冷启动就是其中的一大代表。冷启动的核心是指系统在初始状态下的数据稀缺性，该问题的解决方案有优化模型和增强数据两种。模型对推荐效果的提升远不如增强数据高效便捷，然而对可获得数据的充分建模是不易的，异质的信息并不能通过传统的机器学习或深度神经网络方法进行利用。近些年出现的图神经网络模型给缓解推荐系统的冷启动问题带来新的思路，该方法通过对异质信息的有效建模，成功在计算机视觉，交通预测，分子性质预测，社会计算等诸多领域得到有效应用。

受图神经网络模型的启发，本研究提出一种融合协同信号和商品侧信息的图神经网络推荐算法。所提出的算法在天猫推荐（Rec-Tmall）数据集上进行了 4 项实验，算法 AUC 值平均值约为 0.85，性能相比基于用户的协同过滤推荐算法提升约 32.26%，实验发现选择拼接更新策略和翻译式聚合策略的组合能够使得模型达到最优效果，消融实验也证明了融合商品侧信息能够为模型带来约 10.41%的性能提升。

**关键词：**推荐系统；图神经网络；知识图谱；冷启动

## Abstract

The development of recommendation systems has been plagued by some problems, and cold-start is one of the major representatives. The core of cold-start refers to the scarcity of data in the initial state of the system. There are two solutions to this problem: optimizing the model and enhancing the data. The improvement of the model is far less efficient and convenient than enhancing data. However, it is not easy to fully model the available data. Heterogeneous data cannot be utilized through traditional machine learning or deep neural network methods. In recent years, graph neural network models have brought new ideas to alleviate the cold-start of recommender systems. This method has been successfully applied in many fields such as computer vision, traffic prediction, molecular property prediction, and social computing through effective modeling of heterogeneous data.

Inspired by the graph neural network model, this study proposes a graph neural network recommendation algorithm that integrates collaborative signals and commodity-side information. The proposed algorithm was tested on the Rec-Tmall dataset in 4 experiments. The average AUC value is about 0.85, resulting in about a 32.26% improvement compared to the user-based collaborative filtering recommendation algorithm. It was found that the combination of a concatenation-based update strategy and a TransE-based aggregation strategy can achieve the best effect of the model. The ablation experiment also proved that integrating commodity-side information can bring about a performance improvement of about 10.41% for the model.

**Keywords:** Recommender System, Graph Neural Network, Knowledge Graph, Cold-Start

# 目 录

1 引 言.....	1
1.1 研究背景与意义.....	1
1.1.1 研究背景.....	1
1.1.2 研究意义.....	1
1.2 国内外研究历史与现状.....	2
1.2.1 信息检索.....	2
1.2.2 推荐系统.....	2
1.2.3 图神经网络.....	4
1.2.4 冷启动.....	4
1.3 研究的主要贡献与创新.....	5
1.4 论文的组织结构.....	5
2 相关理论.....	6
2.1 推荐系统.....	6
2.1.1 推荐系统的定义.....	6
2.1.2 推荐算法的分类.....	7
2.2 图神经网络.....	9
2.2.1 图神经网络的定义.....	9
2.2.2 图神经网络的组成.....	9
2.3 知识图谱.....	10
2.3.1 知识图谱的概念.....	10
2.3.2 知识图谱在推荐系统中的应用.....	10
3 融合协同信号和商品侧信息的图神经网络推荐算法.....	11
3.1 数据集和数据预处理.....	12
3.1.1 天猫推荐 (Rec-Tmall) 数据集.....	12
3.1.2 协同信号及知识图谱的构建.....	13
3.2 图神经网络的消息传递策略.....	13
3.2.1 问题描述.....	14
3.2.2 KGCN 聚合策略.....	14
3.2.3 KGCN 更新策略.....	15
3.3 图神经网络的优化策略.....	16
4 实验结果与算法评估.....	16
4.1 实验设定.....	16

4.1.1 评估指标.....	16
4.1.2 实验环境.....	17
4.1.3 超参数设置.....	17
4.2 可行性实验.....	18
4.3 基线对比实验.....	19
4.4 聚合/更新策略对比实验.....	19
4.5 消融实验.....	20
4.6 实验结论.....	21
5 总结与展望.....	21
参考文献.....	22
致 谢.....	26
附录 1 Rec-Tmall 数据集字段含义.....	27
附录 2 模型的 12 个超参数.....	28
附录 3 聚合/更新策略对比实验的实验结果.....	29

# 1 引言

## 1.1 研究背景与意义

### 1.1.1 研究背景

1992年12月，施乐公司开发出了协同过滤推荐算法<sup>[1]</sup>。如果将其视作推荐系统研究的开端，那么该研究领域至今已经发展30年有余。

推荐系统自诞生以来，便逐渐在信息检索领域成为最重要的模型之一，并在社会经济生活当中发挥重要价值。今天我们在互联网的各种平台上听音乐、看视频、浏览新闻、网购商品，都已经离不开推荐系统的“推荐”了。从用户的角度来看，推荐系统帮助人们在海量的信息中找到自己感兴趣的内容，也即有效用的；从平台的角度来看，推荐系统帮助平台运营方和平台的客户推广内容增加营收，也即有效益的。例如，亚马逊网络书城的推荐算法就为亚马逊每年贡献近三十个百分点的创收<sup>[2]</sup>。

从发展阶段来看，推荐系统大致可以划分为浅层模型，神经模型，和图神经网络模型三个阶段<sup>[3]</sup>。今天，得益于充足的计算机算力资源和可获得的大量数据资源，以图神经网络为基础的推荐模型逐渐成为研究热点。与过去的模型只能处理欧几里得空间的数据不同，图神经网络将信息视作节点以及节点间的连接，因此能够更好地利用非欧空间的数据，为用户提供更精准的推荐。

然而，推荐系统本身就面临着用户和物品冷启动的问题，加之图神经网络自身具有的稀疏性也让冷启动问题更加难以解决，因此基于图神经网络解决推荐系统冷启动问题是有研究价值的。推荐系统服务众多领域，包括音乐、视频、新闻、商品等。由于商品推荐领域的数据集最易获得，也最为丰富，因此本研究重点关注于电子商务环境中的商品冷启动问题。

### 1.1.2 研究意义

#### (1) 理论意义

冷启动问题是推荐系统中一个长期存在的问题<sup>[4]</sup>。冷启动问题的具体表现在于，对于新用户来说，由于没有足够的用户信息，系统无法向用户推荐相关的商品；而对于新商品来说，由于没有足够的用户与之互动，商品也很难被系统推荐。因此，对于冷启动问题的研究同样是具有效用和效益的。本研究结合图神经网络为推荐系统的冷启动问题提供新的解决思路，拓展了理论研究的视角。

#### (2) 实践意义

互联网经济的一个典型表现就是平台经济。当今的平台如果想提高用户满意度和产品转化率，就不得不考虑推荐系统等技术的使用。而对于新建立的平台而言，平台中更

多的是新用户和新商品，冷启动问题更加需要被重视。研究冷启动问题对于企业的平台化建设有重要的实践意义和价值。

## 1.2 国内外研究历史与现状

### 1.2.1 信息检索

推荐系统本质上是信息检索的一种方法，本节介绍信息检索的发展以便对推荐系统的外延有更深刻的理解。

信息检索是为了解决信息过载问题的。尽管信息过载的概念直到 1964 年才被首次提出<sup>[5]</sup>，但是信息过载的现象是自人类文明的诞生之日起就有的。随着书籍等人类文明所保存资料的大量积累以及合适的工具技术的出现，信息检索自然应运而生。

- 距今约 2000 年前，我国东汉蔡伦改良造纸术，人类文明产生的信息得以开始被保存，大量信息的产生让早期信息检索技术应运而生——辞书；
- 过去 100 年里，存储技术、数字化技术的诞生让信息得以被数字化地保存，这催生了新方法——电子目录；
- 过去 50 年里，计算机网络技术的飞速发展让信息更海量地出现在人们的视野中，谷歌研发的以 PageRank 为核心的新的信息检索方式<sup>[6]</sup>诞生——搜索引擎；
- 最近的 30 年里，信息指数式爆发，据预测 2025 年人类将产生近 200 万亿 GB 的数据<sup>[7]</sup>，这导致人们更难获取自己需要的信息，也催生了与以往都不同的能够实现“信息找人”的检索技术——推荐系统。
- 今天，信息检索的技术仍然在发展。2022 年末，由 OpenAI 研发的 ChatGPT（人工智能聊天机器人）开放给公众<sup>[8]</sup>，它象征着可自动收集并整理资料的信息检索方式的诞生——对话系统。

辞书，电子目录，搜索引擎，推荐系统和对话系统虽是发展关系，但任何一种方法都未曾消失，本文的核心研究对象推荐系统也不会被对话系统而替代。推荐系统的特点是在潜移默化中学习和理解用户和信息，及其之间的关系，并实现“信息找人”的独特检索方式；智能对话系统则需要用户显式地与之互动，发出提问。这种独特性决定了推荐系统在信息检索领域持续存在的意义。

### 1.2.2 推荐系统

推荐系统发展 30 年来的三个阶段分别是：浅层模型，神经模型，和图神经网络模型。本节对推荐系统三个阶段的模型进行介绍。

#### (1) 浅层模型

浅层模型也可被称为传统模型，其诞生的标志就是 1992 年的协同过滤推荐算法<sup>[1]</sup>。该算法通过分析用户或者物品之间的相似性（“协同”），来预测用户可能感兴趣的内容并将此内容推荐给用户（“过滤”）。按照协同的对象不同，可以将协同过滤分为基于用户

的协同过滤 (User-Based CF) 和基于物品的协同过滤 (Item-Based CF)。协同过滤方法的缺点是需要计算机中存储大型且稀疏的用户-物品共现矩阵, 效率不高同时泛化性不强。因此, 2006 年, Koren 的团队提出矩阵分解技术, 并依靠此方法连续获得两年奈飞推荐大赛的桂冠<sup>[9]</sup>。该方法将共现矩阵通过奇异值分解等方法, 将用户和物品提取为向量, 并通过相似度函数等评分函数预测用户对某物品的评分。不管是协同过滤, 还是矩阵分解, 他们都只能依赖共现矩阵这一单一数据来源, 并未有效利用物品的属性信息或者用户的社交网络信息, 导致推荐不尽人意。因此, 以逻辑回归为基础模型的推荐系统出现, 它能将各类数据输入到模型, 并通过梯度下降的训练方式得到可解释的推荐系统。但是逻辑回归不具备特征交叉的能力, 这导致辛普森悖论<sup>1</sup>类的问题的出现, 因此后续诸多工作开始在特征组合方面做出努力, 比如能够处理二阶特征交叉的 FM<sup>[10]</sup>, 引入特征域概念的 FFM<sup>[11]</sup>, 可以自动化更高阶特征组合的 GBDT+LR<sup>[12]</sup>, 和实现对样本分片的特征组合方法 LS-PLM<sup>[13]</sup>。从协同过滤到 LS-PLM, 这些模型从深度学习的视角来看, 都无非是 2 层及以下的模型, 因此这些传统方法可以被称为浅层模型。

### (2) 神经模型

神经模型是基于深度神经网络的推荐系统模型, 与浅层模型相对应。深度神经网络诞生于 2012 年, 因为极强的拟合能力, 它迅速在计算机视觉、语音处理、自然语言处理等方面得到应用<sup>[14]</sup>。2016 年, 随着微软提出的 Deep Crossing 模型的诞生<sup>[15]</sup>, 深度神经网络也开始正式应用于推荐系统领域。Deep Crossing 模型首先将诸多数据字段进行向量化, 然后将之输入到后续的 3 层深度神经网络中, 端到端地训练用户对物品的评价。该模型能够充分挖掘数据中的价值, 同时能够让数据中的特征得到充分的交叉, 因而能够提供有效的推荐。但是深度网络中的拟合能力过强会削弱模型的“记忆力”, 为了解决这一问题, 谷歌在 2016 年提出 Wide&Deep 模型<sup>[16]</sup>。该模型组合拥有拟合能力强的 Deep 部分和记忆能力强的 Wide 部分。同时, 该模型还推动了神经模型的组合式发展。比如, 组合了注意力模型的 AFM<sup>[17]</sup>, 组合了序列模型的 DIEN<sup>[18]</sup>, 组合了强化学习模型 DRN<sup>[19]</sup>, 以及组合图神经网络模型到推荐系统的相关工作<sup>[20-22]</sup>。

### (3) 图神经网络模型

图神经网络模型本质是一种神经模型, 之所以将其单独地划分一类, 是因为它对非欧几里得数据的利用使其与过去的模型有较大的区别; 同时鉴于本研究重点关注的是图神经网络模型, 因此使用了“浅层模型-深层模型-图神经网络模型”三阶段分类法<sup>[3]</sup>。现有的图神经网络模型可以根据使用的信息类型和推荐任务的不同, 分为用户-物品协同过滤、顺序推荐、社会推荐、基于知识图谱的推荐和其他任务<sup>[23]</sup>。由于本文探讨的模型只涉及用户-物品协同过滤方法和基于知识图谱的方法两种, 因此接下来只对这两种技

---

<sup>1</sup> 辛普森悖论是指, 如果将样本分组, 在组内有优势的某个样本, 可能在取消分组后失去优势。“分组”在推荐系统的视角来看可以理解特征交叉。



术进行介绍。用户-物品协同过滤方法使用共现矩阵的数据构建用户和物品的交互图，并利用聚合和更新的操作不断学习用户和物品的表征，以获得高阶连接性，该方面的典型工作如新加坡国立大学研究团队在 2019 年提出的 NGCF<sup>[21]</sup>。但是，仅利用共现矩阵中的数据过于单薄，基于知识图谱的推荐则可以通过挖掘物品的属性及其之间关系增强物品的表征能力。比如，上海交通大学研究团队在 2019 年提出的 KGCF<sup>[24]</sup>，该模型利用用户和物品的属性信息中的关联性构建知识图谱，并且用多跳邻居的聚合和更新操作捕捉用户潜在的远距离兴趣。

### 1.2.3 图神经网络

随着社会网络、分子结构和知识图谱等海量图谱数据的迅速涌现，近年来，图神经网络的研究逐渐成为深度学习的热点。图神经网络本质上是深度神经网络模型一种自然而然的一般化，它不再局限于欧几里得空间中“横平竖直”的数据，转向对数据和数据之间连接的建模。这种更贴近真实世界的建模让图神经网络具有广阔的应用空间，比如社交网络分析，分子性质预测，知识图谱等。

根据清华大学和其他机构的研究，从图的结构上划分，图神经网络模型可以划分为无向图，有向图，异质图，动态图，超图，有符号图和大规模图<sup>[25]</sup>。根据中国人民大学的研究，从网络结构的不同，图神经网络模型可以划分为图卷积网络，图自编码器，图生成网络，图循环网络和图注意力网络五种<sup>[26]</sup>。根据北京大学和阿里巴巴的研究，从模型使用的聚合和更新函数方面划分，图神经网络模型可以分为 GCN, GraphSAGE, GAT, HetGNN 和 HGNN<sup>[23]</sup>。

由于以上五种聚合和更新函数在推荐系统中均得到应用，因此下面对这些模型进行简要介绍。GCN 是谱分解图卷积模型，它结合了图卷积和神经网络来实现半监督分类的图任务<sup>[27]</sup>。GraphSAGE 从空间图卷积角度建模，对目标节点的邻居进行采样，聚合它们的嵌入，并与目标嵌入合并更新<sup>[28]</sup>。GAT 进一步假设邻居的影响既不完全相同，也不是由图结构预先决定的，因此它通过利用注意力机制来区分邻居的影响，并通过关注其邻居来更新每个节点的向量<sup>[29]</sup>。GGNN 在更新步骤中采用了门控循环单元<sup>[30]</sup>。HGNN 则是一个典型的超图神经网络，它以超图结构编码高阶数据相关性<sup>[31]</sup>。

### 1.2.4 冷启动

推荐系统中的冷启动问题是指，由于新用户或者新物品还未能在系统中产生足够的交互数据，因此新用户难以获得个性化推荐建议、新物品难以得到推荐曝光。从涉及对象来看，冷启动问题可以分为用户冷启动和物品冷启动。而从冷启动地本质——新对象交互数据的稀疏性来看，可以将冷启动工作的研究分为“开源”和“节流”两种解决思路。“开源”是指，既然交互数据稀疏，那么可以从新用户或者新物品的自身特征上挖掘数据来源。比如，Gonzalez Camacho 等人综述了如何利用用户的社交网络信息来增强对

用户的建模<sup>[32]</sup>, FMFC 模型将物品自身带有的属性信息融合进推荐系统中加强了对物品的建模<sup>[33]</sup>, RS-LOD 利用了关联开放数据技术以解决冷启动问题<sup>[34]</sup>。“节流”是指, 既然新对象的相关数据还稀缺, 那么可以构造能更充分利用数据的模型。比如由北京邮电大学等机构的研究团队提出的结合了元学习的 MetaHIN<sup>[35]</sup>。元学习, 是一种“学会学习”的机器学习范式, 它能够快速适应具有稀缺标记数据的新任务, 因而非常适合解决冷启动问题。MetaHIN 通过语义增强的任务构造器和共同适应的元学习器来解决物品冷启动问题。除了元学习, 还有部分工作通过积极学习<sup>[33]</sup>, 局部全体嵌入<sup>[36]</sup>等模型为冷启动问题提供解决思路。

### 1.3 研究的主要贡献与创新

首先, 本研究提出了利用商品侧信息知识图谱进行推荐, 缓解商品推荐冷启动问题的算法模型。其次, 为了验证提出的模型在商品推荐上的性能, 本研究对 Rec-Tmall 数据进行了合理的数据预处理, 构造了可用于知识图谱推荐的数据集。最后, 本研究进行了充分的实验, 通过有效性实验、对比实验及消融实验证明了本研究所提出方法在解决冷启动问题上的作用。

### 1.4 论文的组织结构

本文主要包括五个章节, 其组织结构如图 1-1 所示。

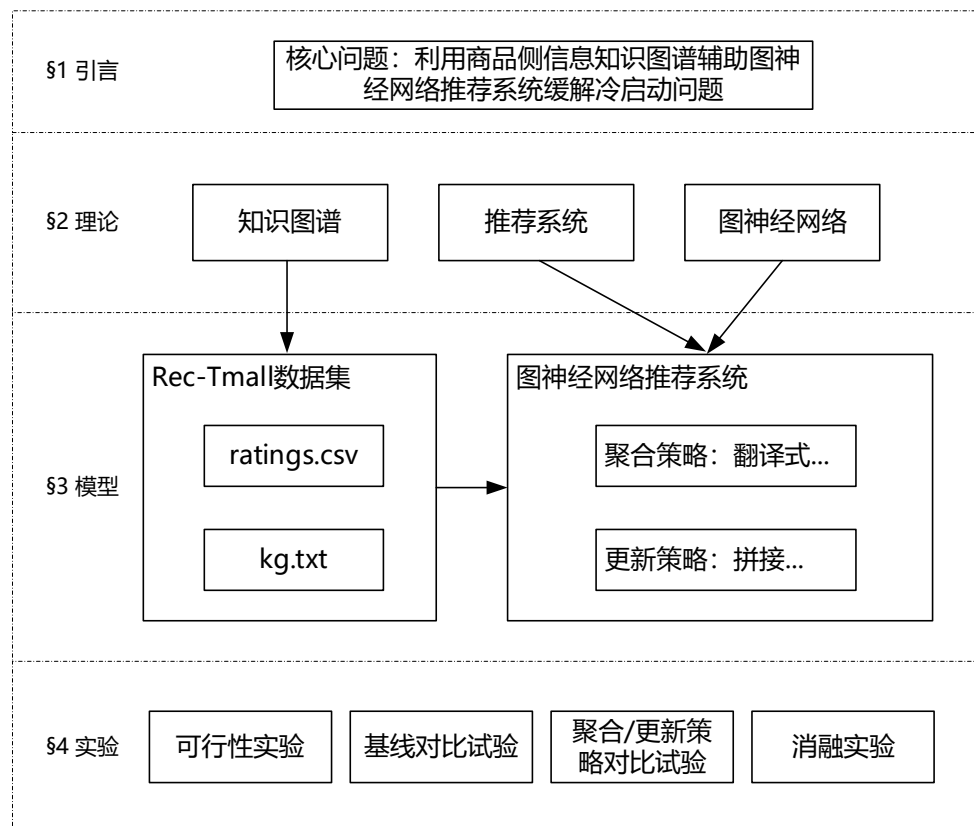


图 1-1 文章组织结构图

第一章为引言。该部分主要介绍本研究开展的背景与意义；从信息检索，推荐系统，图神经网络和冷启动四个方面综述国内外研究历史与现状；概括本研究的主要贡献与创新；以及介绍论文的组织结构。

第二章为相关理论。该部分重点介绍与本研究提出的算法相关的三种重要技术的基本原理，它们分别是推荐系统，图神经网络和知识图谱。

第三章为提出的方法。该部分从数据、模型和优化三方面展开介绍本研究提出的融合协同信号和商品侧信息的图神经网络推荐算法。数据上重点讲图的构建；模型上重点讲节点的聚合和更新策略；优化上重点讲损失函数的设计。

第四章为实验结果与算法评估。该部分介绍本研究的实验设定以及所进行的四项实验及其实验结果，这四项实验包括可行性实验、基线对比实验、聚合/更新策略对比试验及消融实验。

第五章为总结与展望。该部分总结了研究的贡献与不足，对未来的研究方向作出展望。

## 2 相关理论

本章重点关注三个研究主题，它们是推荐系统，图神经网络和知识图谱。其中，推荐系统是本文重点探讨的研究对象；而图神经网络由于能够充分利用异质的信息，为更精准推荐提供可能；知识图谱则是承载异质信息的载体，是直接解决推荐系统冷启动问题的关键。

### 2.1 推荐系统

#### 2.1.1 推荐系统的定义

采用 Resnick 等在 1997 给出的非形式化定义，推荐系统（Recommender System, RS）是指“利用电子商务网站向客户提供商品信息和建议，帮助用户决定应该购买什么产品，模拟销售人员帮助客户完成购买过程”<sup>[4]</sup>。这个定义不免过时，因为今天推荐的物品不仅可以是商品，也可以是音乐、电影等。因此综合多篇文献与书籍的观点，本文对推荐系统作出通用的定义，如定义 2-1 所示。

**定义 2-1** 推荐系统是指利用特定算法预测（predicate）用户（user）可能喜欢的物品（item），并向用户推荐（recommend）有限个物品的，一般由第三方平台开发的信息管理系统。

一方面，定义 2-1 完整地体现出推荐系统要解决的两个问题，分别是“预测（用户可能喜欢哪些物品）”和“推荐（物品很多，推荐哪些）”。另一方面，定义 2-1 也体现出推荐的三个组成要素和背后的三方参与者。推荐系统的三个组成要素分别是：用户，物

品和推荐算法。这三个要素都对应于一种参与者：它们分别是用户，客户和平台<sup>[3]</sup>，如图 2-1 所示。

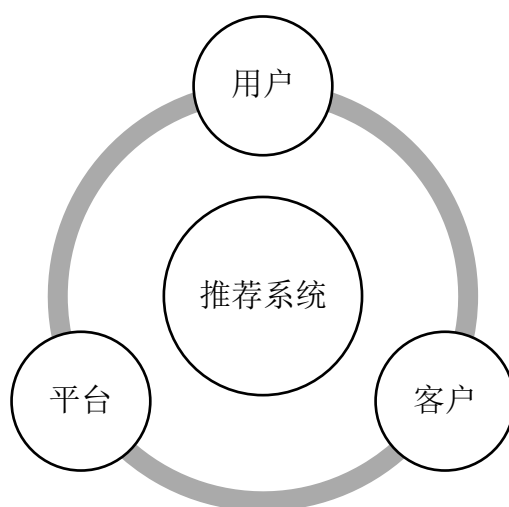


图 2-1 推荐系统的三方参与者

任何一个推荐系统想要进行完美的推荐，就必须兼顾用户、客户和平台三方的利益。以电子商务平台，亚马逊的在线商城为例，用户即消费者，客户即第三方卖家，而平台则是亚马逊自身。如果亚马逊想做一个完美的推荐系统，就必须同时满足消费者（用户）可以高效地发现自己喜欢的商品，卖家（客户）的商品总是有机会且有效地被推荐给消费者，而亚马逊（平台）则要不断通过推荐系统收集高质量的消费者数据以提高系统效率并增加营收。

在学术界，上面所讨论的非形式化的定义并没有达成共识，而形式化的定义则早就在学界有定论。Adomavicius 等在 2005 年将推荐系统定义如下<sup>[5]</sup>（见定义 2-2）。可以发现，这个定义只解决了“预测”，而没有解决“推荐”。事实上，这也是因为准确的预测是个性化推荐的基础，也是核心。本文也仅专注于预测问题的解决。

**定义 2-2** 推荐系统被定义为要完成如下任务：设  $C$  是所有用户的集合， $S$  是所有可以推荐给用户的物品的集合，设效用函数  $u(\cdot)$  可以计算物品  $s$  对用户  $c$  的效用，即  $u: C \times S \rightarrow R$ ，其中  $R$  是一定范围内的非负实数，则推荐要解决的问题就是找到使得效用  $u$  最大的那些物品  $s^*$ 。如式 2-1 所示。

$$\forall c \in C, s^* = \arg \max_{s \in S} u(c, s) \quad (2-1)$$

### 2.1.2 推荐算法的分类

推荐算法是推荐系统的核心，而推荐算法能够实现精准推荐依赖于其中的精准“预测”，即计算物品对用户的效用。本节对这类用于预测的排序层推荐算法的分类进行简单介绍，不涉及召回层的算法。从大类来看，排序层推荐算法主要分为浅层模型和深层

模型两种，如图 2-2 所示。

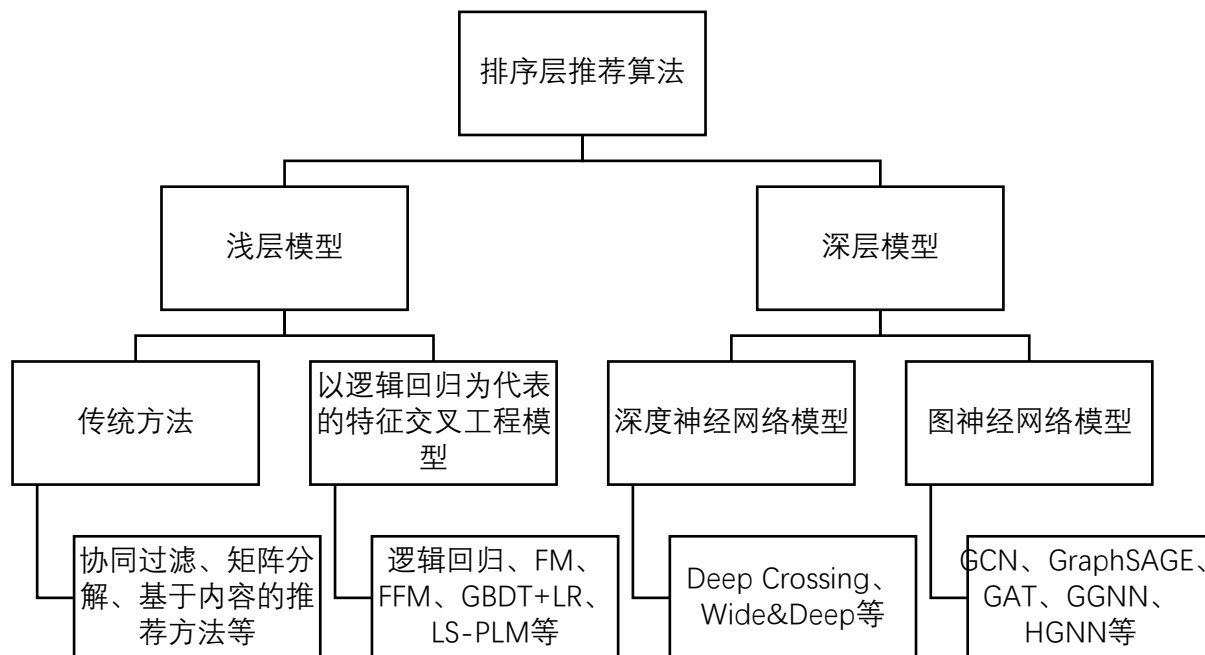


图 2-2 排序层推荐算法的分类

首先是浅层模型，这部分模型的发展历史大致从 1992 年到 2016 年。浅层模型包括传统方法和以逻辑回归为代表的机器学习方法。前者根据主流分类方法<sup>[5,7]</sup>又可细分为协同过滤，基于内容的推荐和混合型推荐。后者则包括为了改进逻辑回归所进行的一系列特征交叉工程的模型，如 FM、FFM、GBDT+LR、LS-PLM 等。这些以特征交叉工程为主的模型在引言章节都已有所阐述，下面仅对传统方法展开阐述。

### (1) 协同过滤

协同过滤(Collaborative Filtering, CF)是推荐系统中广泛使用的最经典的技术之一。该技术通过分析用户或者物品之间的相似性(“协同”),来预测用户可能感兴趣的内容并将此内容推荐给用户<sup>[10]</sup>。该方法又可分为基于用户的协同过滤(User-Based CF)和基于物品的协同过滤(Item-Based CF)。

### (2) 基于内容的推荐

基于内容的推荐(Content-Based Recommendation)根据物品的信息(颜色、描述、价格等)、用户的信息(年龄、职业、学历等)及用户对物品的操作(评价、浏览时间、购买次数等)来构建推荐算法模型,为用户提供推荐服务。该类推荐算法具体包括信息检索领域知名的 TF-IDF 模型、概率论相关的贝叶斯分类器等。

### (3) 混合型推荐

一般来说,混合型推荐(Hybrid Recommendation)是通过各种组合方式将协同过滤和基于内容的推荐两种方法混合在一起的推荐方式,它可以在一定程度上避免单一算法

的缺陷。

而对于深层模型，则是以神经网络为代表的各类模型。从本研究的视角来看，深层模型包括传统的深度神经网络模型和图神经网络模型，这些模型也都在前言部分有所阐述，这里不再赘述。

## 2.2 图神经网络

### 2.2.1 图神经网络的定义

Sanchez-Lengeling 等在 2021 年为图神经网络定义如下<sup>[37]</sup>（见定义 2-3）。图神经网络采用的是图到图（graph-in, graph-out）的架构，其输入和输出的图在连接性上保持置换不变性（permutation invariance），即图在空间中的连接点数量不变而且连接点位不发生改变。在保证这样的架构约束下，图神经网络利用“消息传递神经网络（message passing neural network）”框架来改变图所有属性本身的表征，例如节点的权重。图 2-3 展示了这一框架对图所作出的变换：从  $n$  到  $n+1$  的变换，节点之间的连接性不变，而节点的权重（节点边框粗细）和边的权重（边的粗细）有变化。

**定义 2-3** 图神经网络（Graph Neural Network, GNN）就是能够在保持图的置换不变性的同时，对图的所有属性（这些属性包括节点、边和图自身。）进行的优化变换。

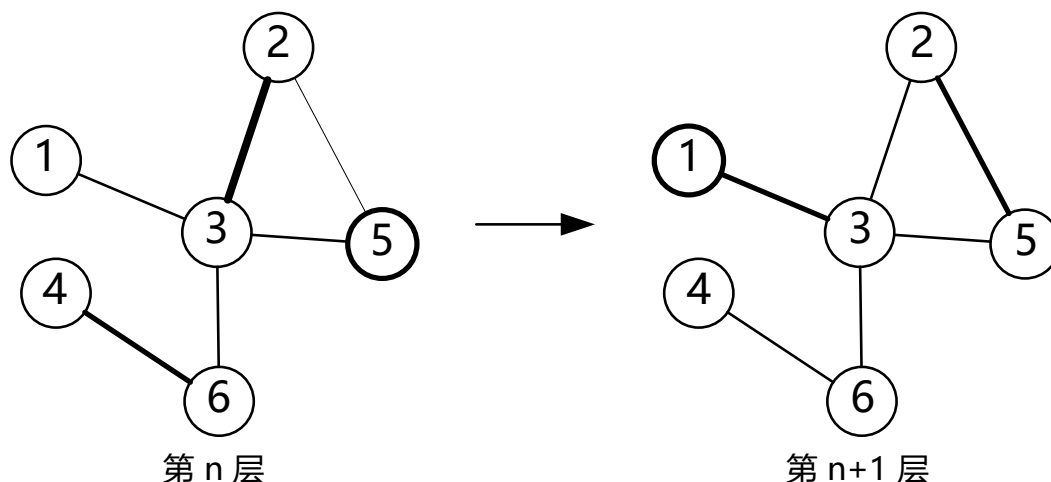


图 2-3 图神经网络的优化示意图

### 2.2.2 图神经网络的组成

从图神经网络的架构来分析，一个图神经网络模型的构建需要考虑两个关键问题，一是输入的图结构如何从现实生活中构建？二是从输入到输出的中间层的变换如何实现？对于第一个问题，传统算法提供诸多方案，最简单的是邻接矩阵方法，它将用户与物品的交互置于一张二维表中，如表 2-1 所示。对于第二个问题，刚刚所提到的“消息传递神经网络”框架则是经典的解决方案之一。

表 2-1

四位用户对四部电影的评价矩阵

	鬼吹灯	小鬼当家	千与千寻	天气之子
小刚	0.8	0.2	N/A	0.1
小红	0.2	N/A	0.7	0.9
小明	N/A	0.6	N/A	N/A
小芳	N/A	0.3	0.4	0.2

图神经网络的消息传递框架能够在保证不改变图的连接性的同时实现图的优化变换。消息传递框架由简单的两个步骤组成，即“聚合周围的特征”和“更新当前中心节点”两步，分别可以由式 2-2 和式 2-3 表示。

$$n_v^{(l)} = \text{Aggregator}_l(\{h_u^{(l)}, \forall u \in \mathcal{N}_v\}) \quad (2-2)$$

$$h_v^{(l+1)} = \text{Updater}_l(\{h_v^{(l)}, n_v^{(l)}\}) \quad (2-3)$$

其中， $h_u^{(l)}$  是节点  $u$  在第  $l$  层的向量化表征， $u \in \mathcal{N}_v$  代表  $u$  是  $v$  的邻域节点，而  $\text{Aggregator}_l$  和  $\text{Updater}_l$  分别是聚合函数和更新函数。对于不同的图神经网络模型来说，其  $\mathcal{N}_v$ ， $\text{Aggregator}_l$  和  $\text{Updater}_l$  都有不同的具体表现，一个常见的设定是： $\mathcal{N}_v$  即一跳邻域， $\text{Aggregator}_l$  即平均池化操作， $\text{Updater}_l$  即向量拼接。

## 2.3 知识图谱

### 2.3.1 知识图谱的概念

知识图谱（Knowledge graph, KG）最初是由谷歌公司于 2012 提出的<sup>[38]</sup>。自此，许多公司和组织都开始建立自己的知识图谱，例如 YAGO、Freebase、Wikidata 和 DBpedia、CN-DBpedia 等。虽然知识图谱已经发展了近十年，但它仍然没有一个明确的定义。一个被普遍接受的对知识图谱的认识是<sup>[39]</sup>，它是一种旨在累积和传达真实世界知识的数据图，其节点代表真实世界中的实体，而边代表这些实体之间的多种多样的关系。知识图谱可以用三元组的集合来表示：

$$\mathcal{G} = \{(h, r, t)\} \quad (2-4)$$

其中， $h$  和  $t$  分别代表头实体和尾实体， $r$  代表头尾实体之间的关系。例如，一个关于首都的知识图谱可能是： $\{(Beijing, \text{capital\_of}, China), (Tokyo, \text{capital\_of}, Japan), \dots\}$ 。

### 2.3.2 知识图谱在推荐系统中的应用

浅层模型时代大多数模型仅考虑用户-物品交互图，即用户对物品产生点击关系、购买关系、评论关系的图，而由于交互图本身具有很大的稀疏性，因造成冷启动问题的出

现。后来逐渐有研究将物品的属性信息也融入到推荐系统中，例如显式地给出物品的价格、流行度等信息；也有将用户的社交关系信息融入到推荐系统中的，因为社交关系对用户的消费行为具有显然的影响，这些信息的融入有效地缓解了冷启动的问题。将物品的属性信息和用户的社交关系信息融入到推荐系统中的方法有很多种，浅层模型时代的逻辑回归模型就是其中之一，但其用户对物品效用的预测受制于层数限制而拟合效果不优；在深层模型的今天，将这些信息建模为知识图谱（如图 2-4 所示），并借助图神经网络模型进行优化，最后利用一定的向量间评分函数来预测效用值，就能够达到充分利用信息解决冷启动问题并提供精准推荐的目的。下面给出一些三元组的例子，来展示如何对这些信息进行建模。

在建模用户-物品协同信号（即用户物品交互信息）时，一个对应的知识图谱可能是{(Emily Brown,bought,iPhone), (William John,like,Google Pixel), ...}.

在建模用户侧信息（即社交关系信息）时，一个对应的知识图谱可能是{(Emily Brown,friend\_of,William Johnson), (William John,son\_of,Ashley John), ...}.

在建模物品侧信息（物品属性信息）时，一个对应的知识图谱可能是{(iOS,os\_of,iPhone), (Android,os\_of,Google Pixel) ...}.

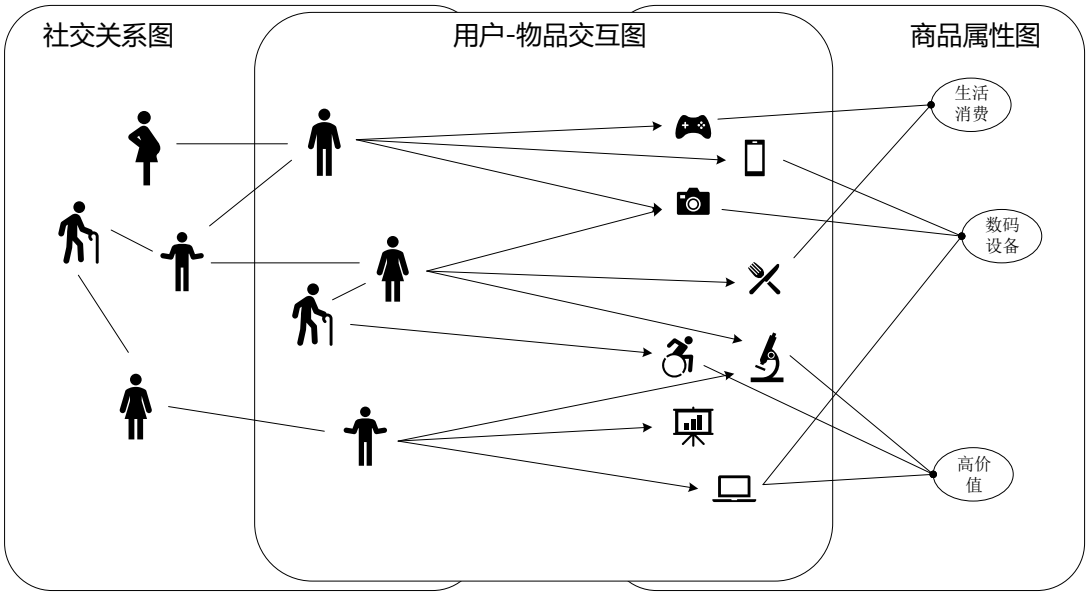


图 2-4 推荐系统可以利用的各类信息所组成的图结构

### 3 融合协同信号和商品侧信息的图神经网络推荐算法

本章详细阐述提出的融合协同信号和商品侧信息的图神经网络推荐算法。该算法使用天猫的推荐系统数据集，数据集包含了用户-物品协同信号和大量可供提取商品侧信息；接着，通过对天猫数据集的预处理得到推荐系统的输入端知识图谱；应用 KGCN 模



型提出的消息传递策略给出图神经网络的消息传递策略和优化策略；在卷积神经网络迭代给定步数后，计算用户对物品的效用得分。算法的框架如图 3-1 所示。

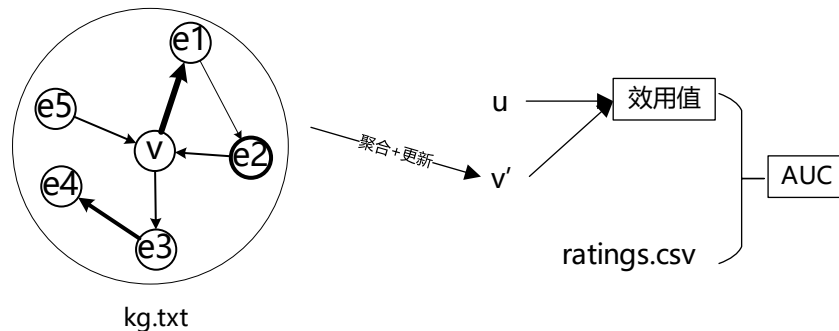


图 3-1 算法框架

### 3.1 数据集和数据预处理

#### 3.1.1 天猫推荐 (Rec-Tmall) 数据集

Rec-Tmall<sup>[40,41]</sup>数据集是由天猫提供的推荐系统评测数据集，目前在天池数据集网站上开放下载使用<sup>2</sup>。该数据集由样本集，和全量集两个部分组成，他们的描述统计信息如表 3-1 所示。样本集供快速原型开发使用，全量集供标准的测试使用。两个数据集均由三部分构成，分别是商品数据，日志数据和评论数据。

表 3-1 Rec-Tmall 数据集基本数据统计

	用户数	商品数	交互数
样本集	3,429	562	4,844
全量集	9,774,184	8,133,507	1,333,729,303

商品数据文件存储着商品侧信息，其每一行包括 6 个字段，分别是商品的 ID，商品标题，商品图片对应的链接，商品的类别，商品的品牌和商品的销售者。其中商品的类别由“父类别-子类别”的二级分类体系表示。日志数据文件中存储着用户-商品交互信息，其每一行包括 4 个字段，分别是商品 ID，用户 ID，用户行为，和产生行为时对应的时间戳。其中用户行为字段有四个可能的值，分别是点击、收藏、添加至购物车和购买（按照行为代表的用户对商品的青睐程度升序排列）。评论数据文件仍然存储着用户-商品交互信息，其每一行包括 3 个字段，分别是商品 ID，用户 ID，和用户对商品的评论。详细的数据集各字段及其含义见附录 1。

本研究未使用全量集，仅使用样本集作为实验数据集。此外，尽管评论数据文件中

<sup>2</sup> Rec-Tmall 数据集访问下载地址：<https://tianchi.aliyun.com/dataset/140281>

蕴藏了丰富的用户-商品交互信息，但由于评论文本是自然语言文本，将其转化为知识图谱有赖于自然语言处理的方法和技术，如知识提取技术。这并非本文所研究的重点，因此本研究不使用评论数据文件。综上，本文所用到的数据是样本集中的商品数据和日志数据。

### 3.1.2 协同信号及知识图谱的构建

在得到了天猫推荐数据集的原始数据文件后，首先进行数据的预处理，对协同信号和知识图谱建模。

**用户-物品协同信号建模** 日志数据文件中包含了用户的行为数据。用户的行为作为一种隐式反馈数据（即计算机所不能有效利用的“点击”“收藏”等标签），不能被推荐系统所直接使用。本研究合理假设“用户对商品的行为与用户对商品的评分有正向相关性”并建立行为到评分的映射，如表 3-2 所示。另外，由于该研究仅考虑静态的推荐，而不考虑在线的序列式的推荐，遂不使用用户产生行为对应的时间戳信息。

表 3-2 行为到评分的映射关系假设

	点击	收藏	入购物车	购买
对应评分	1.25	2.5	3.75	5

**商品侧信息建模** 商品数据文件使用除商品图片链接外的其余 5 个字段。根据这些字段的语义，对于每个观测值，本研究都提取出 4 种预定义的三元组，如表 3-3 所示。这 4 种三元组共同构成商品侧信息的知识图谱。

表 3-3 人工预定义的 4 种类型的三元组

头实体	关系	尾实体
商品 ID (item_id)	属于 (belong_to)	父类别-子类别 (category)
父类别-子类别 (category)	属于 (belong_to)	父类别(category 前半部分)
商品 ID (item_id)	商品品牌 (product_brand)	品牌 (brand_id)
商品 ID (item_id)	商品商家 (selled_by)	商家 (seller)

## 3.2 图神经网络的消息传递策略

KGCN<sup>[24]</sup> (Knowledge Graph Convolution Network) 模型是最早提出的面向推荐系统的，在知识图谱上进行消息传递的图神经网络框架。它在知识图谱上进行卷积迭代，联合协同信号计算用户对物品的效用得分，这是一种直观、简洁的解决思路。本研究采用 KGCN 框架<sup>3</sup>来进行前文提到的知识图谱上的消息传递，而在具体的聚合策略上，本文

<sup>3</sup> 本研究选择使用 PyTorch 版本的 KGCN 代码作为基础并加以修改，代码的使用和修改获得了作者的授权。原作者

与 KGCN 有所不同。

### 3.2.1 问题描述

在协同信号方面，给定  $m$  个用户  $\mathcal{U} = \{u_1, u_2, \dots, u_m\}$  以及  $n$  件商品  $\mathcal{V} = \{v_1, v_2, \dots, v_n\}$ 。这些用户和物品之间的协同信号，首先根据行为到评分的映射由隐式反馈转换为显示反馈，接着被建模为矩阵  $Y \in \mathbb{R}^{m \times n}$ ，其中物品  $v$  对用户  $u$  的效用即  $y_{uv} \in \{1.25, 2.5, 3.75, 5\}$ 。商品侧信息方面，对表 3-3 中的实体和关系编码并训练嵌入。这些训练出的向量表示构成商品侧信息知识图谱  $\mathcal{G} = \{(h, r, t)\}$ ，其中  $h, t \in \mathcal{E}, r \in \mathcal{R}$ 。而  $\mathcal{E}$  和  $\mathcal{R}$  分别代表实体集和关系集。

表 3-4 重要符号说明

符号	含义
$m, n$	用户的数量，商品的数量
$\mathcal{U}, \mathcal{V}, \mathcal{E}, \mathcal{R}$	用户集合，商品集合，实体集合，关系集合
$u, v, e, h, t, r$	用户，物品，实体，头实体，尾实体，关系
$Y, y_{uv}, \hat{y}_{uv}$	用户-物品协同信号矩阵，效用，效用预测值
$\mathcal{G}, (h, r, t)$	商品侧信息知识图谱，三元组
$\mathcal{F}(\cdot), \Theta, W, b$	模型函数，及模型的参数
$\mathcal{N}(v)$	物品 $v$ 的一跳邻域节点集合
$\mathcal{S}(v)$	从 $\mathcal{N}(v)$ 中抽取的固定大小的抽样集合
$v_{\mathcal{N}(v)}^u$	考虑用户 $u$ 影响，物品 $v$ 的聚合结果
$v_{\mathcal{N}(v)}$	不考虑用户 $u$ 影响，物品 $v$ 的聚合结果

在给定用户-物品协同信号矩阵  $Y$  以及商品侧信息知识图谱  $\mathcal{G}$  后，模型需要根据这些数据预测用户  $u$  是否对他之前没有交互过的项目  $v$  有潜在兴趣。因此，模型的目标是预测用户  $u$  对商品  $v$  效用值，如式 3-1 所示。此外，表 3-4 给出了本模型使用到的所有符号的说明。

$$\hat{y}_{uv} = \mathcal{F}(u, v | Y, \mathcal{G}; \Theta) \quad (3-1)$$

### 3.2.2 KGCN 聚合策略

在聚合方面，KGCN 模型提出的是关系敏感的一跳邻域聚合策略，这是一种注意力

代码见 <https://github.com/zzaebok/KGCN-pytorch>，授权信见 <https://github.com/zzaebok/KGCN-pytorch/issues/8>。

机制的具体应用。它利用用户对不同关系的注意力不同，而对其邻域的节点赋予不同的权值，如式 3-2 所示。其中， $\hat{\pi}_{r_{v,e}}^u$  是关系  $r_{v,e}$  对用户  $u$  的注意力得分，具体计算方法见式 3-3。

$$v_{\mathcal{N}(v)}^u = \sum_{e \in \mathcal{N}(v)} \hat{\pi}_{r_{v,e}}^u e \quad (3-2)$$

$$\hat{\pi}_{r_{v,e}}^u = \frac{\exp(\pi_{r_{v,e}}^u)}{\sum_{e \in \mathcal{N}(v)} \exp(\pi_{r_{v,e}}^u)} \quad (3-3)$$

尽管 KGCN 提出的聚合策略考虑到具有不同关系的实体之间的影响，但这样的聚合策略有三方面的缺陷。一是对异质信息的建模是间接的，影响中心节点的本质上是头节点和和中心节点的相似性，而并非通过边相连的尾实体。二是其计算复杂度较高，从式 3-2 和式 3-3 可知， $v_{\mathcal{N}(v)}^u$  的计算复杂度至少为  $O(n^2)$ 。三是它在聚合的过程中考虑用户  $u$  的影响，本研究认为用户对物品的影响是主观的，这里应当是明确地计算物品聚合的客观信息。

为了解决上述缺陷，本文尝试以 TransE<sup>[42]</sup>为启发，聚合经过“翻译 ( $h = t - r$ )”的头实体，如式 3-4 所示。基于 TransE 的聚合器计算复杂度低至  $O(n)$ ，并且后续的实验中表明，它与 KGCN 的注意力模型相比，具有更高的预测精准度。此外，也减少了用户的影响。

$$v_{\mathcal{N}(v)} = \sum_{e \in \mathcal{N}(v)} (e - r_{v,e}) \quad (3-4)$$

### 3.2.3 KGCN 更新策略

在更新方面，KGCN 的提出者提供了三种具体的方法，它们分别是求和，拼接和直接使用邻域表征。求和策略计算原表征和聚合后的表征的向量和，并施加线性变换和激活函数，如式 3-5 所示；拼接策略将原表征和聚合后的表征首尾相连，并施加线性变换和激活函数，如式 3-6 所示；直接使用邻域表征策略则如字面意思，忽略原表征，直接在聚合后的表征上施加线性变换和激活函数，如式 3-7 所示。这些方法复杂度基本一致，并且是有效且通用的一些策略。本研究直接探究使用这三种更新策略对消息传播的影响，这将在实验阶段进行对比分析。

$$\text{upd}_{\text{sum}} = \sigma(W \cdot (v + v_{\mathcal{S}(v)}^u) + b) \quad (3-5)$$

$$\text{upd}_{\text{concat}} = \sigma(W \cdot \text{concat}(v, v_{\mathcal{S}(v)}^u) + b) \quad (3-6)$$

$$\text{upd}_{neighbor} = \sigma(W \cdot v_{S(v)}^u + b) \quad (3-7)$$

### 3.3 图神经网络的优化策略

在优化策略上，本研究使用 KGCN 框架中提出的带有负采样和 L2 正则化的损失函数，如式 3-8 所示。负采样使得训练更加高效，而 L2 正则化则确保模型不易过拟合。

$$\mathcal{L} = \sum_{u \in \mathcal{U}} \left( \sum_{v: y_{uv}=1} \mathcal{J}(y_{uv}, \hat{y}_{uv}) - \sum_{i=1}^{T^u} \mathbb{E}_{v_i \sim P(v_i)} (\mathcal{J}(y_{uv_i}, \hat{y}_{uv_i})) \right) + \lambda \|\Theta\|_2^2 \quad (3-8)$$

其中， $\mathcal{J}$  是交叉熵损失， $P$  是负采样分布函数， $T^u$  是用户  $u$  的负采样个数。而  $T^u = |\{v: y_{uv} = 0\}|$  和  $P$  服从均匀分布。最后一项， $\lambda \|\Theta\|_2^2$  则是 L2 正则化项。

## 4 实验结果与算法评估

### 4.1 实验设定

本研究将物品被点击的概率（即点击率）作为预测目标。在点击率预测中，本实验使用测试集中的每个用户与物品之间的交互作为参考，应用训练好的模型来预测用户对物品点击的概率，并将实际值与预测值比对评估。本研究的所有代码和实验均可以在 <https://github.com/Ki-Seki/KGCN-pytorch-updated/tree/ugt-only> 中找到。

#### 4.1.1 评估指标

本实验使用 ROC AUC 来评估预测效果。ROC (receiver operating characteristic curve) 是指受试者工作特征曲线，它被广泛应用于在给定的阈值下，对阴性/阳性分类效果的评测当中，其参数方程如式 4-1 所示。以本研究为例，ROC 可以对模型分类点击/不点击的效果进行合理评估。而 AUC (area under curve) 是在全体阈值情况（即  $[0,1]$ ）下求取 ROC 的积分值，该值越大则表明，模型在大多数阈值下的分类效果越好，其计算方法如式 4-2 所示。

$$\begin{cases} y(\theta) = TPR = \frac{TP}{TP + FN} \\ x(\theta) = FPR = \frac{FP}{FP + TN} \end{cases} \quad (4-1)$$

$$AUC = \int_0^1 y(\theta) d\theta \quad (4-2)$$

其中， $TP, FP, TN, FN$  分别代表真阳性，假阳性，真阴性和假阴性， $R$  代表比率， $\theta$  是阈值。 $TP, FP, TN, FN, TPR, FPR$  均是  $\theta$  的函数，但为了表示的简洁，公式中没有

加上。

#### 4.1.2 实验环境

硬件条件上，本实验运行在带有 GPU 加速的个人笔记本电脑上；软件条件方面，本实验主要使用 PyTorch 深度学习开发框架。硬件和软件的细节如表 4-1 和表 4-2 所示。

表 4-1 硬件环境细节

项目	值
处理器	Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz 2.11 GHz
内存	16GB
硬盘	512GB
显卡	NVIDIA GeForce MX250
专用显存	2GB

表 4-2 软件环境细节

项目	值
操作系统	Windows 11 家庭中文版 22H2
虚拟机	WSL 1.1.6.0
编程语言	Python 3.10.9
深度学习框架	PyTorch 1.13.1

#### 4.1.3 超参数设置

本研究提供的可供调节的实验超参数共 12 个，其中对接下来的实验具有直接影响的参数共 4 个，它们是 aggregator, mixer, n\_epochs 和 random\_seed. 其中，aggregator 与 mixer 控制聚合和更新策略，n\_epochs 控制迭代次数，random\_seed 控制实验的随机数生成器以保证实验的可复现性。这 4 个超参数的详细解释如表 4-3 所示，完整的 12 个超参数解释如附录 2 所示。

表 4-3 对实验具有直接影响的 4 个超参数及其解释

超参数	取值	解释
aggregator	sum, concat 或 neighbor	三种更新策略：求和，拼接和邻域
mixer	transe 或 attention	两种聚合策略：翻译式和注意力式
n_epochs	正整数	迭代次数，保证拟合效果
random_seed	任意数字	指定随机数种子，控制变量，保证可复现性

## 4.2 可行性实验

为了验证模型的可行性，本研究选择在 3 个不同随机数种子的设定下进行了 3 次实验，实验结果如表 4-4 所示。三次实验的 AUC 值均超过 0.8，平均值约为 0.85，模型效果优秀。

表 4-4 可行性实验的结果<sup>4</sup>

随机数种子	实验结果（收敛点, AUC）
6666	<b>140, 0.844176</b>
59	<u>143, 0.820407</u>
123	144, <b>0.898722</b>

此外，从实验指标的迭代过程（如图 4-1）上来看，模型均在至少前 120 次迭代时，AUC 值就超过了 0.8；测试集上达到拟合最优点均在至少前 100 次时；三次实验总用时 73.1 秒，平均用时约 24.4 秒，每一个迭代用时约 0.2 秒，算法效率良好。总体上，模型的时间复杂度，收敛性良好。

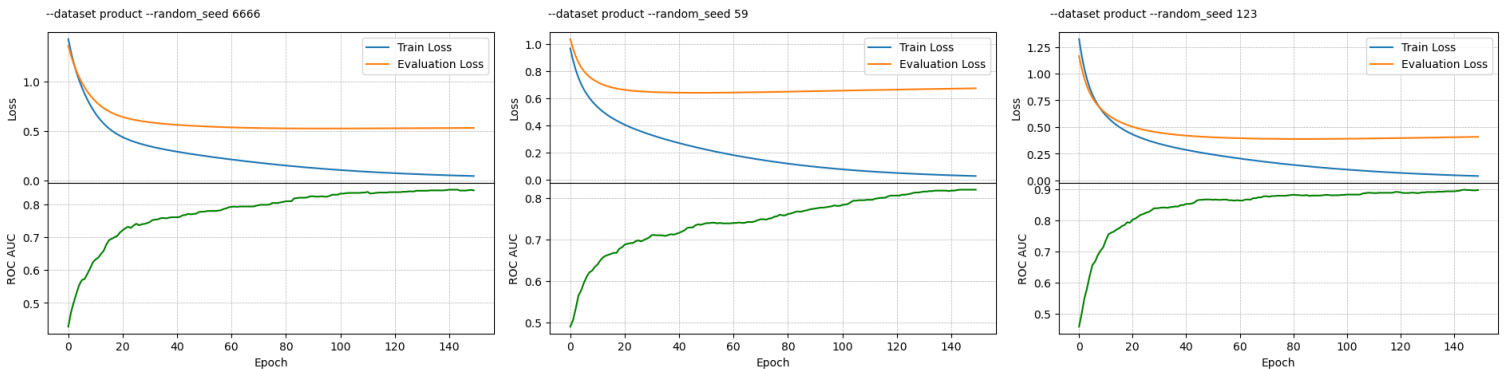


图 4-1 可行性实验指标迭代图示

为了更直观地观察模型的预测效果，这里展示第一次实验中的第一批次中的前 8 组点击率预测结果和实际点击与否的值，如表 4-5 所示。可以发现除第四对和第六对匹配度不好，其余均匹配良好。

表 4-5 第一批次中前 8 组点击率的预测值和实际值

预测值	0.4755	0.9410	0.8889	0.9919	0.8850	0.1131	0.8510	0.1162
实际值	0	1	1	0	1	1	1	0

<sup>4</sup> 实验结果中的最优值用加粗格式表示，次优值用下划线表示，下同。

### 4.3 基线对比实验

所提出的模型不仅需要有效，而且必须相对传统方法具有性能上的优越性。本研究选择基于用户的协同过滤推荐算法作为实验的基线算法，在 5 个不同随机数种子的设定下进行了 5 次实验，实验结果如表 4-6 所示。

表 4-6 基线对比实验的结果（基线侧）

随机数种子	实验结果（AUC）
1	<b>0.616041</b>
999	0.533585
345	0.553429
124	0.526271
3345	<u>0.594942</u>

五次实验的 AUC 值均值约为 0.56，最大值约为 0.62，其性能均小于本研究所提出模型的性能。可行性实验中显示，所提出模型的 AUC 最小值约为 0.82，相较基线模型至少有 32.26% 的性能提升。实验结果表明，本研究所提出模型相较传统模型有一定的优越性。

### 4.4 聚合/更新策略对比实验

不同的聚合与更新策略对于图神经网络的迭代有着直接的影响，本研究对文中提到的 3 种更新策略与 2 种聚合策略两两结合进行对比实验，实验结果如表 4-7 所示。

可以发现，本研究提出的翻译式聚合策略相较于注意力式策略而言，均具有非常显著的性能提升，AUC 值平均提升 6.67%，并且保持了与注意力式几乎相同的收敛速度。在更新策略方面，拼接更新策略相比另外两种有显著优势，邻域更新策略次之，而求和策略则最次。

表 4-7 聚合/更新策略对比实验的结果

更新策略（参数名）	聚合策略（参数名）	实验结果（收敛点, AUC）
求和（sum）	翻译式（transe）	149, 0.831771
	注意力式（attention）	149, 0.775663
拼接（concat）	翻译式（transe）	148, <b>0.856771</b>
	注意力式（attention）	<b>122</b> , 0.824716
邻域（neighbor）	翻译式（transe）	<u>144</u> , <u>0.840483</u>
	注意力式（attention）	148, 0.771828

实验结果表明，选择拼接更新策略和翻译式聚合策略的组合能够使得模型达到最优



效果。在这样的组合策略下，实验指标迭代情况如图 4-2 所示。完整的实验结果与指标迭代图示见附录 3。

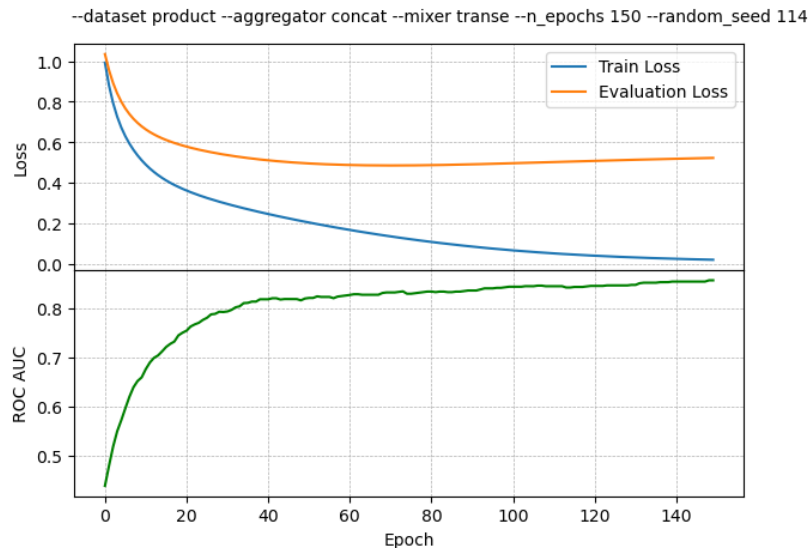


图 4-2 拼接更新策略和翻译式聚合策略组合下的指标迭代图

#### 4.5 消融实验

本模型能提高推荐效果的最基本假设即，增加商品侧知识图谱信息能够缓解推荐系统冷启动问题，提升模型推荐精准度。为了验证这一假设，本研究进行了消融实验，分别在相同情况下，进行使用和不使用知识图谱的消融实验。消融实验在相同情况下进行了两次，如表 4-8 所示。实验时将构造好的知识图谱文件手动修改<sup>5</sup>，以模拟知识图谱不存在时的推荐效果。

表 4-8 两次消融实验的结果

实验（随机数种子）	情形	实验结果（收敛点, AUC）
第一次（1）	使用知识图谱	146, <b>0.830729</b>
	不使用知识图谱	<b>134</b> , 0.794886
第二次（71）	使用知识图谱	146, <b>0.818182</b>
	不使用知识图谱	<b>138</b> , 0.703456

从实验结果可以看出，使用知识图谱对性能的提升是显然的，两次实验性能平均提升为 10.41%。但是收敛性上，不使用知识图谱的情形总是表现更好，这是由牺牲性能换来的。但是对于使用知识图谱的情形来说，平均牺牲 10 次的迭代，换取平均 10.41%的

<sup>5</sup> 将知识图谱文件中第一组三元组保留，其余删除。该做法能最便捷地模拟知识图谱不存在的情况。

性能提升是值得的。

#### 4.6 实验结论

以上实验充分证明了模型在提供有效推荐上的可行性，以及该模型与基于用户的协同过滤算法相比的优越性，发现了选择拼接更新策略和翻译式聚合策略的组合能够使得模型达到最优效果，证明了所利用的知识图谱能为模型性能的提升带来帮助，缓解推荐系统的冷启动问题。

## 5 总结与展望

本研究提出了融合协同信号和商品侧信息的图神经网络推荐算法，该算法通过知识图谱建模商品信息，利用图神经网络进行推荐，有效缓解了推荐系统中的冷启动问题。为了验证该算法在商品推荐方面的有效性及优越性，本研究根据 Rec-Tmall 数据集构建了知识图谱推荐数据集，并在该数据集上进行了充分的实验，包括可行性实验、基线对比实验、聚合/更新策略对比实验和消融实验四项。实验表明本方法在解决冷启动问题上确有优势。

然而本研究仍存在一些待解决的问题。第一，由于实验环境的限制，本研究未能使用全量数据集进行实验，因此对模型在工业界应用的可行性检验不足。第二，本研究对模型的各方面极限的实验不够完整，如并未检验图神经网络的迭代层数的极限。第三，建模商品侧信息可以缓解冷启动问题，但从用户侧建模社交关系信息也能达到同样目的，而本研究在这方面未能做出分析对比。第四，本研究所进行的实验有赖于“行为能代表评分”的假设，而该假设的合理性仍需深入讨论和分析。第五，在实验评估指标方面，本研究仅考虑了 AUC 这一指标，较为单一。

## 参考文献

- [1] Goldberg D, Nichols D, Oki B M, et al. Using collaborative filtering to weave an information tapestry[J]. Communications of the ACM, 1992, 35(12): 61-70.
- [2] 项亮. 推荐系统实践[M]. 人民邮电出版社, 2012.
- [3] Gao C, Zheng Y, LI N, et al. A Survey of Graph Neural Networks for Recommender Systems: Challenges, Methods, and Directions[M]. arXiv, 2022.
- [4] Gope J, Jain S K. A survey on solving cold start problem in recommender systems[C]//2017 International Conference on Computing, Communication and Automation (ICCCA). Greater Noida: IEEE, 2017: 133-138.
- [5] Gross B M. The Managing of Organizations: The Administrative Struggle[M]. Free Press of Glencoe, 1964.
- [6] Page L, Brin S, Motwani R, et al. The PageRank Citation Ranking : Bringing Order to the Web[C]//The Web Conference. 1999.
- [7] Reinsel D, Gantz J, Rydning J. Data Age 2025[R]. International Data Corporation (IDC), 2017.
- [8] OpenAI. ChatGPT: Optimizing Language Models for Dialogue[EB]//OpenAI. (2022-11-30).
- [9] Koren Y, Bell R, Volinsky C. Matrix Factorization Techniques for Recommender Systems[J]. Computer, 2009, 42(8): 30-37.
- [10] Rendle S. Factorization Machines[C]//2010 IEEE International Conference on Data Mining. 2010: 995-1000.
- [11] Juan Y, Zhuang Y, Chin W S, et al. Field-aware Factorization Machines for CTR Prediction[C]//Proceedings of the 10th ACM Conference on Recommender Systems. New York, NY, USA: Association for Computing Machinery, 2016: 43-50.
- [12] He X, Pan J, Jin O, et al. Practical Lessons from Predicting Clicks on Ads at Facebook[C]//Proceedings of the Eighth International Workshop on Data Mining for Online Advertising. New York, NY, USA: Association for Computing Machinery, 2014: 1-9.
- [13] Gai K, Zhu X, Li H, et al. Learning Piece-wise Linear Models from Large Scale Data for Ad Click Prediction[M]. arXiv, 2017.
- [14] 王喆. 深度学习推荐系统[M]. 电子工业出版社, 2020.
- [15] Shan Y, Hoens T R, Jiao J, et al. Deep Crossing: Web-Scale Modeling without Manually Crafted Combinatorial Features[C]//Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, NY, USA: Association

- for Computing Machinery, 2016: 255-262.
- [16] Cheng H T, Koc L, Harmsen J, et al. Wide & Deep Learning for Recommender Systems[C]//Proceedings of the 1st Workshop on Deep Learning for Recommender Systems. New York, NY, USA: Association for Computing Machinery, 2016: 7-10.
- [17] Xiao J, Ye H, He X, et al. Attentional Factorization Machines: Learning the Weight of Feature Interactions via Attention Networks[M]. arXiv, 2017.
- [18] Zhou G, Mou N, Fan Y, et al. Deep Interest Evolution Network for Click-Through Rate Prediction[C]//Proceedings of the AAAI Conference on Artificial Intelligence: vol. 33. 2019: 5941-5948.
- [19] Zheng G, Zhang F, Zheng Z, et al. DRN: A Deep Reinforcement Learning Framework for News Recommendation[C]//Proceedings of the 2018 World Wide Web Conference. Republic and Canton of Geneva, CHE: International World Wide Web Conferences Steering Committee, 2018: 167-176.
- [20] Ying R, He R, Chen K, et al. Graph Convolutional Neural Networks for Web-Scale Recommender Systems[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York, NY, USA: Association for Computing Machinery, 2018: 974-983.
- [21] Wang X, He X, Wang M, et al. Neural Graph Collaborative Filtering[C]//Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, NY, USA: Association for Computing Machinery, 2019: 165-174.
- [22] He X, Deng K, Wang X, et al. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation[C]//Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York, NY, USA: Association for Computing Machinery, 2020: 639-648.
- [23] Wu S, Sun F, Zhang W, et al. Graph Neural Networks in Recommender Systems: A Survey[J]. ACM Computing Surveys, 2023, 55(5): 1-37.
- [24] Wang H, Zhao M, Xie X, et al. Knowledge Graph Convolutional Networks for Recommender Systems[C]//The World Wide Web Conference. New York, NY, USA: Association for Computing Machinery, 2019: 3307-3313.
- [25] Zhou J, Cui G, Hu S, et al. Graph neural networks: A review of methods and applications[J]. AI Open, 2020, 1: 57-81.
- [26] 吴博, 梁循, 张树森, 等. 图神经网络前沿进展与应用[J]. 计算机学报, 2022, 45(1): 35-68.

- [27] Kipf T N, Welling M. Semi-Supervised Classification with Graph Convolutional Networks[C]//International Conference on Learning Representations. 2017.
- [28] Hamilton W, Ying Z, Leskovec J. Inductive Representation Learning on Large Graphs[C]//Advances in Neural Information Processing Systems: vol. 30. Curran Associates, Inc., 2017.
- [29] Veličković P, Cucurull G, Casanova A, et al. Graph Attention Networks[C]//International Conference on Learning Representations. 2018.
- [30] Li Y, Tarlow D, Brockschmidt M, et al. Gated Graph Sequence Neural Networks[M]. arXiv, 2017.
- [31] Feng Y, You H, Zhang Z, et al. Hypergraph Neural Networks[C]//Proceedings of the AAAI Conference on Artificial Intelligence: vol 33. 2019: 3558-3565.
- [32] Gonzalez Camacho L A, Alves-Souza S N. Social network data to alleviate cold-start in recommender system: A systematic review[J]. Information Processing & Management, 2018, 54(4): 529-544.
- [33] Zhu Y, Lin J, He S, et al. Addressing the Item Cold-Start Problem by Attribute-Driven Active Learning[J]. IEEE Transactions on Knowledge and Data Engineering, 2020, 32(4): 631-644.
- [34] Natarajan S, Vairavasundaram S, Natarajan S, et al. Resolving data sparsity and cold start problem in collaborative filtering recommender system using Linked Open Data[J]. Expert Systems with Applications, 2020, 149: 113248.
- [35] Lu Y, 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. Virtual Event CA USA: ACM, 2020: 1563-1573.
- [36] Saveski M, Mantrach A. Item cold-start recommendations: learning local collective embeddings[C]//Proceedings of the 8th ACM Conference on Recommender systems - RecSys '14. Foster City, Silicon Valley, California, USA: ACM Press, 2014: 89-96.
- [37] Sanchez-Lengeling B, Reif E, Pearce A, et al. A Gentle Introduction to Graph Neural Networks[J]. Distill, 2021, 6(9): e33.
- [38] Singhal A. Introducing the Knowledge Graph: things, not strings[EB]//Google. (2012-05-16).
- [39] Hogan A, Blomqvist E, Cochez M, et al. Knowledge Graphs[J]. ACM Computing Surveys, 2021, 54(4): 71:1-71:37.
- [40] Zhang Y, Pang L, Shi L, et al. Large Scale Purchase Prediction with Historical User Actions on B2C Online Retail Platform[M]. arXiv, 2015.

- [41] Zhong W, Jin R, Yang C, et al. Stock Constrained Recommendation in Tmall[C]//Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, NY, USA: Association for Computing Machinery, 2015: 2287-2296.
- [42] Bordes A, Usunier N, Garcia-Duran A, et al. Translating Embeddings for Modeling Multi-relational Data[C]//Advances in Neural Information Processing Systems 26 (NIPS 2013). 2013.

## 致 谢

在本文的撰写过程中，我有许多要感谢的人：感谢潘勇教授对我论文的悉心指导与督促；感谢我的学长余同学为我认真审校了文章；感谢我的师姐牛同学为我的选题提供的帮助。不仅仅是他们，在我的大学四年生活当中还有许多要感谢的人。他们是陪伴我、支持我的女朋友小崔同学，教会我勇敢的板友郭同学，一起闲聊的胡老师，领我入门科研的史老师，提升我英语能力的刘老师和 Elt，默默关心我的常老师，带给我诸多帮助的刘老师、汪老师和潘老师等等。除此之外，我的成长还少不了我家人的支持。我的爸爸和妈妈是我能平心静气地学习的最大保障；我的妹妹是我自驱力的来源之一，我希望能成为她的榜样。我的大学过得并没有很累，我能悠游自得且相对安心快乐地做着想做的事情，朝着自己的理想前进。这样的生活方式得以实现也必须要感谢我自己，感谢自己永不气馁地大胆尝试，感谢自己总是打破上限地积极探索！

## 附录 1 Rec-Tmall 数据集字段含义

表 1-1 商品数据文件

字段	含义
Item_id	在 [1,8133507] 范围内的整数，表示一个唯一的商品。由多于两个商家同时提供的商品会在多行中以不同的 ID 记录。
Title	包含多个关键词的字符串，由空格分隔。这些词是由自然语言处理系统从原始标题中提取出来的。
Pict_url	指向相应在线图片的链接。
Category	字符串，格式为“x-y”。其中“x”表示父类别，“y”表示子类别。
Brand_id	形如“b1”，“b89366”等的字符串，表示商品的品牌。
Seller_id	形如“s1”，“s86799”等的字符串，表示销售该商品的卖家。

表 1-2 日志数据文件

字段	含义
Item_id	在 [1,8133507] 范围内的整数，表示一个唯一的商品。
User_id	形如“u9774184”等的字符串，表示一个唯一的用户。
Action	表示行为类型的字符串，如“click”，“collect”，“cart”，“alipay”，分别表示“点击”，“收藏”，“入购物车”和“购买”。
Vtime	行为的时间戳，格式为“yyyy-mm-dd hh:mm:ss”。

表 1-3 评论数据文件

字段	含义
Item_id	在 [1,8133507] 范围内的整数，表示一个唯一的商品。
User_id	形如“u9774184”等的字符串，表示一个唯一的用户。
feedback	包含多个关键词的字符串，由空格分隔。这些词是由自然语言处理系统从原始评论中提取出来的。
rate_pic_url	一个链接到相应在线图片的 URL。
Gmt_create	评论的时间戳，格式为“yyyy-mm-dd hh:mm:ss”。



## 附录 2 模型的 12 个超参数

表 2-1

模型的 12 个超参数及其解释

超参数	取值	解释
aggregator	sum, concat, neighbor	三种更新策略：求和，拼接和邻域
mixer	transe, attention	两种聚合策略：翻译式和注意力式
n_epochs	任意正整数	迭代次数，保证拟合效果
random_seed	任意数字	指定随机数种子，保证可复现性
dataset	product, music, movie	三个内置的数据集，本研究仅用 product
neighbor_sample_size	任意正整数	邻域固定抽样大小，默认值 8
dim	任意正整数	用户和实体向量表征的维度，默认值 16
n_iter	任意正整数	计算实体表征时的迭代次数，默认值 1
batch_size	任意正整数	计算一批样本的数量，默认值 32
l2_weight	任意正实数	L2 正则化项权重，即 $\lambda$ , 默认值 $10^{-4}$
lr	任意 $[0,1]$ 内实数	学习率，默认值 $5^{-4}$
ratio	任意 $[0,1]$ 内实数	训练集所占大小的比率，默认值 0.8

### 附录 3 聚合/更新策略对比实验的实验结果

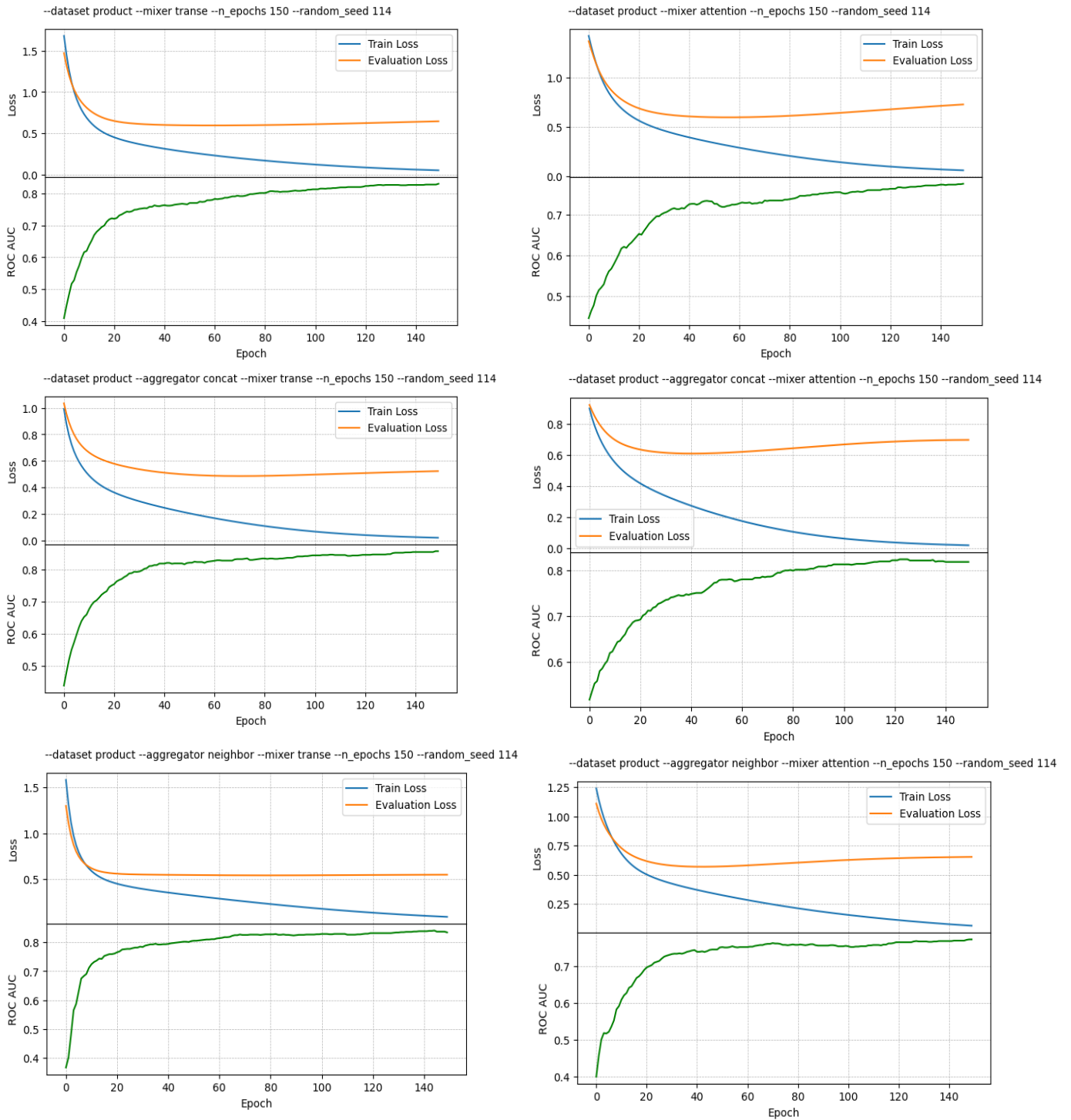


图 3-1 实验的指标迭代图示<sup>6</sup>

<sup>6</sup> 总计 6 次实验，横向上，其聚合策略依次是翻译式和注意力式；纵向上，其更新策略依次是求和，拼接和直接使用邻域表征。

## 毕业论文（设计）独创性声明

本人郑重声明：所呈交的毕业论文（设计）是本人在指导教师的指导下，独立进行研究取得的成果。除文中已经注明引用的内容外，本论文不包含其他个人或集体已经发表或撰写的研究成果，也不包含为获得河南财经政法大学或其他教育机构的学位或证书所使用过的材料。因本文引起的法律后果完全由本人承担。

签名：

日期：

## 毕业论文（设计）使用授权声明

本人完全了解河南财经政法大学有关使用毕业论文（设计）的规定，即：学校有权保留所送交的毕业论文（设计），允许毕业论文（设计）被查询和借阅；可以公布毕业论文（设计）内容及可采用影印、缩印或其他复制手段保存毕业论文（设计）。

学生签名：

导师签名：

日期：