

基于隐式反馈的 Top-10 推荐列表预测

傅显坤

March 6, 2024

Abstract

在选择众多的互联网上，需要对相关信息进行过滤、优先排序和高效传递，以缓解信息过载问题。推荐系统通过搜索大量动态生成的信息来解决信息过载的问题，从而为用户提供个性化的内容和服务。本文基于老师给出的用户隐式反馈的商品数据集，探讨了六种不同的推荐系统算法模型的原理，并结合 QRec 项目使用 python 对这六种算法模型进行了代码的实现。通过对比六种算法模型在验证数据集上的算法表现，选择出表现最好的 LightGCN 算法模型作为最终的预测算法模型。最后给出使用 LightGCN 算法模型对测试集内的用户进行 top-10 预测的结果。

1 Introduction

可用数字信息量和 Internet 访问者数量的爆炸式增长带来了信息过载的潜在挑战，这阻碍了用户及时访问 Internet 上感兴趣的项目。信息检索系统，如 Google、DevilFinder 和 Altavista 已经部分解决了这个问题，但信息的优先级和个性化（系统将可用内容映射到用户的兴趣和偏好）却不存在。这比以往任何时候都增加了对推荐系统的需求。推荐系统是处理信息过载问题的信息过滤系统根据用户的偏好、兴趣或观察到的关于项目的行为，从大量动态生成的信息中过滤出重要的信息片段。推荐系统能够根据用户的个人资料预测特定用户是否喜欢某个项目。

本文在老师给出的用户隐式反馈的商品数据集上面，基于[GitHub 上开源项目 QRec](#)，探讨了业界应用广泛且已经比较成熟的 Neural Graph Collaborative Filtering (NGCF) [1], LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation (LightGCN) [2], Dual Channel Hypergraph Collaborative Filtering (DHCF) [3], Adversarial Personalized Ranking for Recommendation (APR) [4], Collaborative Denoising Auto-Encoders for Top-N Recommender Systems (CDAE) [5], Neural Collaborative Filtering [6] 的原理和实现方法，以及对映的在验证数据集上的算法表现。最终选择在验证集上表现最优的 LightGCN 算法模型对测试集内的用户进行 top-10 预测。

本文在 section 2 详细介绍了推荐系统的原理、方法和评估的相关先导知识。并对于推荐系统中隐式反馈数据集进行详细的介绍和描述。并解释了本文为什么要用到的 NGCF, LightGCN, DHCF, APR, CDAE, NeuMF 六种算法模型对用户进行预测。

在 section 3 中详细介绍了图神经网络 GNN 在推荐系统领域的应用，并详细介绍了 Neural Graph Collaborative Filtering (NGCF) [1], LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation (LightGCN) [2], Dual Channel Hypergraph Collaborative Filtering (DHCF) [3] 三种算法的原理。

在 section 4 中详细介绍了深度神经网络 DNN 在推荐系统领域的应用，并详细介绍了 Adversarial Personalized Ranking for Recommendation (APR) [4], Collaborative Denoising Auto-Encoders

for Top-N Recommender Systems(CDAE)[5],Neural Collaborative Filtering[6] 三种算法的原理。

在 section5中结合[GitHub 上开源项目 QRec](#)对六种算法模型进行 Python 代码实现并给出六种算法模型的训练过程和并对六种算法模型在验证集上的结果进行了详细的对比，并选择最终在验证集上表现最好的 LightGCN 算法，对测试集内的用户进行 top-10 预测。

2 推荐系统

2.1 推荐系统的原理、方法和评估

在阅读了 Recommendation systems: Principles, methods and evaluation 论文的基础上 [7],总结了推荐系统的原理、方法和评估。推荐系统被定义为复杂信息环境下用户的一种决策策略。推荐系统从电子商务的角度被定义为一种帮助用户搜索与用户兴趣和偏好相关的知识记录的工具。推荐系统被定义为在没有足够的个人知识或替代方案经验的情况下，协助和增强使用他人推荐做出选择的社会过程的一种手段。推荐系统通过向用户提供个性化、专有的内容和服务推荐来解决用户通常遇到的信息过载问题。

2.1.1 推荐过程的阶段

推荐系统的阶段分为信息收集阶段(Information collection phase),学习阶段 (Learning phase)和预测/推荐阶段 (Prediction/Recommendation phase)。信息收集阶段：收集用户的相关信息，为预测任务生成用户画像或模型，包括用户的属性、行为或用户访问的资源内容。学习阶段：它应用学习算法从信息收集阶段收集的反馈中过滤和利用用户的特征。预测/推荐阶段：它推荐或预测用户可能喜欢什么样的项目。对于推荐系统各个阶段的关系描述见图1。

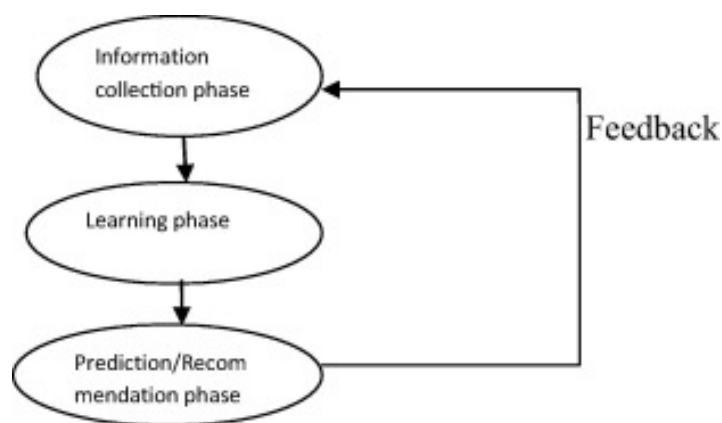


Figure 1: Recommendation phases.

2.1.2 推荐过滤技术

推荐系统的技术主要是推荐过滤技术，同时可以分为基于内容的过滤 (Content-base filtering technique)，协同过滤 (Collaborative filtering technique)，混合过滤 (Hybrid filtering technique)，详细的解释见图2。基于内容的过滤：基于内容的技术是一种领域相关的算法，它更强调分析项目的属性以生成预测。常用语推荐网页、出版物和新闻等文件。协同过滤：协同过滤是一种独立于领域的预测技术，用于无法通过电影和音乐等元数据轻松且充分描述的内容。协同过滤技术通过建

立用户对项目的偏好的数据库（用户-项目矩阵）来工作。然后，它通过计算个人资料之间的相似性来匹配具有相关兴趣和偏好的用户，以提出建议。这样的用户建立了一个名为邻居的组。用户会得到对那些他以前没有评价过但已经被他附近的用户正面评价过的项目的推荐。混合过滤：混合过滤技术结合不同的推荐技术以获得更好的系统优化，避免纯推荐系统的一些局限性和问题。混合技术背后的想法是，算法的组合将提供比单一算法更准确和有效的推荐，因为另一种算法可以克服一种算法的缺点。使用多种推荐技术可以抑制组合模型中单个技术的弱点。方法的组合可以通过以下任何一种方式完成：算法的单独实现和结果的组合，在协作方法中使用一些基于内容的过滤，在基于内容的方法中使用一些协作过滤，创建一个统一的推荐系统，带来两种方法一起使用。

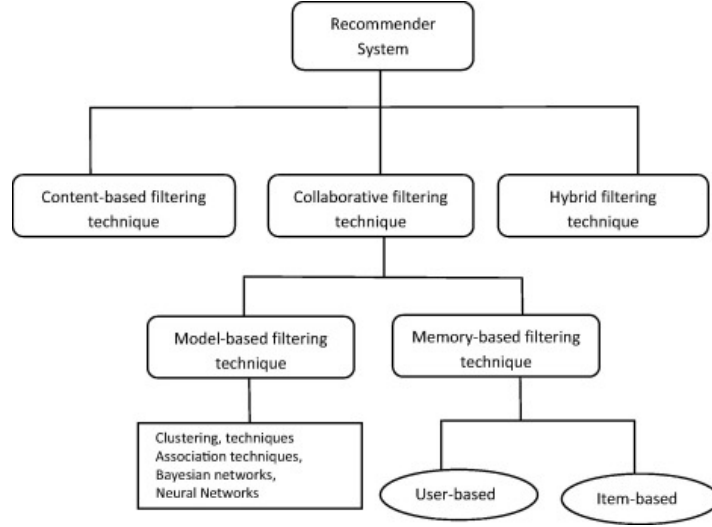


Figure 2: Recommendation techniques.

2.1.3 推荐系统的算法评估指标

推荐算法的质量可以用不同类型的衡量标准来评估，这些标准可以是准确性或覆盖率。使用的指标类型取决于过滤技术的类型。准确性是指在所有可能的推荐中正确推荐的比例，而覆盖率则是指系统能够提供推荐的搜索空间中的对象的比例。衡量推荐过滤系统准确性的指标分为统计和决策支持准确性指标。每个指标的适用性取决于数据集的特点和推荐系统要做的任务类型。

统计准确性指标通过将预测评级直接与实际用户评级进行比较来评估过滤技术的准确性。Mean Absolute Error(MAE)、Root Mean Square Error (RMSE) 和相关性通常用作统计精度指标。MAE 是最流行和最常用的，它是衡量推荐偏离用户特定值的指标。它的计算方式如下

$$MAE = \frac{1}{N} \sum_{u,i} |p_{u,i} - r_{u,i}|.$$

其中 $p_{u,i}$ 代表用户 u 对于项目 i 的预测评分， $r_{u,i}$ 是实际评分， N 是项目集的评分总数。MAE 越低，推荐系统预测用户评分就越准确。此外，RMSE 是：

$$RMSE = \sqrt{\frac{1}{N} \sum_{u,i} (p_{u,i} - r_{u,i})^2}.$$

均方根误差 (RMSE) 更强调绝对误差越大，RMSE 越低，推荐准确度越好。

常用的**决策支持准确度**指标有逆转率、加权误差、接收者操作特征（ROC）和精确召回曲线（PRC）、精确率、召回率、F-measure 和 NDCG。这些指标帮助用户从可用的项目中选择质量非常高的项目。这些指标将预测程序视为二进制操作，将好的项目与不好的项目区分开来。在对一些特定算法的性能进行综合评估时，ROC 曲线非常成功。精确度是指推荐项目中与用户实际相关的部分，而召回率可以定义为相关项目中也是推荐项目集的一部分的部分。它们的计算方法是：

$$Precision = \frac{Correctlyrecommendeditems}{Totalrecommendeditems}$$

$$Recall = \frac{Correctlyrecommendeditems}{Totalusefulrecommendeditems}.$$

下面定义的 F 度量有助于将精度和召回率简化为单个指标。结果值使得算法之间和跨数据集的比较变得非常简单和直接：

$$F - measure = \frac{2PR}{P + R}.$$

NDCG (normalized discounted cumulative gain)：NDCG 是衡量排名系统有效性的指标，考虑了相关项目在排名列表中的位置。它基于这样的想法，即排名较高的项目应该比排名较低的项目获得更多的信用。NDCG 的计算方法是将排名列表的折扣累积增益 (DCG) 除以理想排名列表的 DCG，理想排名列表是相关项目按最佳顺序排列的列表。NDCG 的取值范围为 0 到 1，值越高表示性能越好。公式表示如下图3。

$$NDCG@K = \frac{DCG@K}{IDCG@K} = \frac{\sum_{i=1}^{k(actual\ order)} \frac{Gains}{\log_2(i+1)}}{\sum_{i=1}^{k(ideal\ order)} \frac{Gains}{\log_2(i+1)}}$$

Figure 3: Here is the formula for NDCG@K.

在算法实现过程中对 $MAE, RMSE, Precision, Recall, F - measure$ 和 $NDCG$ 分别进行了实现，以便对算法进行准确的评估。

2.2 数据推荐中的隐式反馈

隐式反馈是指用户在进行操作或行为时不需要明确提供喜好或评分，而系统可以通过这些隐式行为来推断用户的偏好。相比显式反馈（如评分或喜好标签），隐式反馈更加隐含和间接，但它在大数据推荐系统中扮演着重要的角色。

隐式反馈数据可以包括以下几种类型：

1. 点击行为：当用户在推荐系统中点击某个物品时，这可以被视为对该物品的兴趣和偏好的一种暗示。

2. 购买行为：当用户购买某个物品时，这也暗示了用户对该物品的偏好。
3. 浏览行为：用户在系统中浏览某个物品的行为，可以作为对该物品感兴趣程度的一种暗示。
4. 收藏行为：当用户将某个物品收藏到个人收藏夹中时，这表明用户对该物品有一定的喜好。
5. 分享行为：用户在社交媒体或其他渠道上分享某个物品时，这可能暗示了用户对该物品的兴趣和推荐价值。

本次实验用到的是用户对于商品的点击行为

利用隐式反馈数据进行推荐有以下优点：1. 用户无需明确评分，减轻了用户负担，提高了用户体验。2. 隐式反馈数据通常更易获取，且存在丰富的隐式信号可以用于推断用户的兴趣和偏好。3. 隐式反馈数据具有更高的实时性，能够快速反映用户的最新行为。

然而，隐式反馈数据也存在一些挑战：1. 隐式反馈数据往往是稀疏的，因为用户只会与少数物品进行交互，导致数据稀疏性问题。2. 隐式反馈数据隐含用户的偏好，但不能提供具体的用户偏好程度，缺乏明确的评分指标。3. 隐式反馈数据存在一定的噪声和歧义性，需要进行合理的建模和处理。

针对隐式反馈数据的特点和挑战，图神经网络（GNN）在大数据推荐系统中的应用提供了一种强大的建模框架。通过利用图神经网络，可以有效地捕捉用户与物品之间的复杂关系，挖掘潜在的兴趣和推荐信号。这种基于图的建模方式可以更好地处理隐式反馈数据的稀疏性和噪声问题，并提高推荐系统的准确性。所以本文选取了三种图神经网络模型（NGCF, LightGCN, DHCF）进行实现，并对用户 top-10 商品进行推荐预测。除了图神经网络之外，深度神经网络（DNN）在也提供了一种强大的建模框架。通过利用深度神经网络，可以对用户和物品的关系进行深度学习，从而提高推荐系统的性能和准确性。深度神经网络在大数据推荐系统中的应用具有以下优势：通过多层的隐藏层和激活函数，深度神经网络能够提供更丰富的表达能力，从而对用户行为进行更精细的建模。所以本文也选取了三种深度神经网络模型（APR, CDAE, NeuMF）进行实现，并对用户 top-10 商品进行推荐预测。最后对六种算法进行对比分析。

3 图神经网络 GNN

在阅读了《Graph Neural Networks in Recommender Systems:A Survey》论文后，我对图神经网络在推荐系统中的应用和前沿技术有了一个大概的了解 [8]。这篇论文由北大及阿里巴巴整合 211 篇论文后于 2022 年五月发表在 ACM Computing Surveys 上，该论文阐述了近些年较火的图神经网络 (Graph Neural Network,GNN) 在推荐系统中的应用综述。

如今，深度学习模型已成为学术研究和工业应用中推荐系统的主要方法，因为它能够有效地捕捉非线性和非平凡的用户-项目关系，并轻松整合丰富的数据源，例如上下文、文本和视觉信息。在所有这些深度学习算法中，一种是基于图学习的方法，它从图的角度考虑推荐系统中的信息。推荐系统中的大多数数据本质上都具有图结构。例如，推荐应用程序中的交互数据可以由用户和项目节点之间的二分图表示，观察到的交互由链接表示。甚至用户行为序列中的项目转换也可以构建为图形。当结合结构化外部信息（例如用户之间的社会关系）时，将推荐制定为图形任务的好处变得尤为明显以及与项目相关的知识图谱。通过这种方式，图学习为推荐系统中丰富的异构数据建模提供了一个统一的视角。基于图学习的推荐系统的早期工作利用图嵌入技术对节点之间的关系进行建模，可以进一步分为基于分解的方法、基于分布式表示的方法和基于神经嵌入的方法。受 GNN 在学习图结构数据方面的卓越能力的启发，最近出现了大量基于 GNN 的推荐模型。

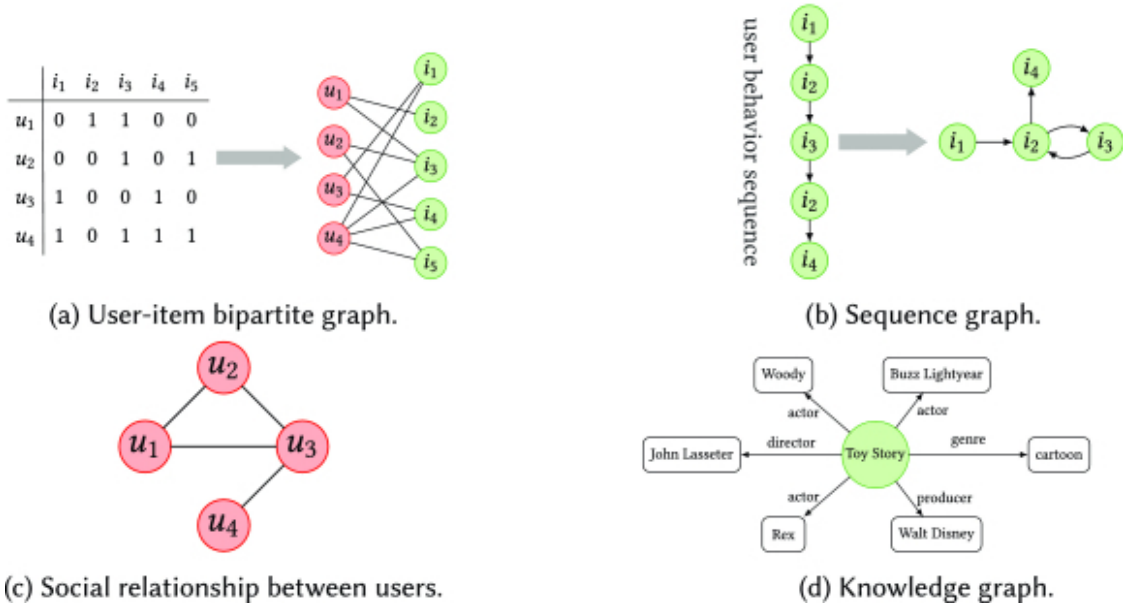


Figure 4: Representative graph structures in recommender systems.

GNN 能够广泛且深入的应用于推荐系统领域最直观的原因是，GNN 技术已被证明在各个领域的图数据表示学习方面具有强大的功能，并且推荐中的大多数数据本质上具有如图4所示的图结构。例如，用户-项目交互数据可以由用户和项目节点之间的二分图（如图4 (a) 所示）表示，其中链接表示相应用户和项目之间的交互。此外，可以将一系列项目转换为序列图，其中每个项目都可以与一个或多个后续项目相关联。图4(b) 显示了序列图的示例，其中连续项之间存在边。与原始序列数据相比，序列图允许更灵活地处理项目到项目的关系。除此之外，一些附加信息 (side information) 也自然具有图结构，例如社会关系和知识图谱，如图4 (c) 和4 (d) 所示。

在本文中由于数据集是基于隐式反馈的，并且并没有用户之间的社交网络。所以我阅读了 NGCF, LightGCN 和 DHCF 的实现原理。本部分接下来将对这三种模型进行详细的解释。

3.1 NGCF 算法原理和实现

在阅读了《Neural graph collaborative filtering》[1] 论文的基础上，整理了如下知识。

学习用户和项目的向量表示（又名嵌入）是现代推荐系统的核心。从早期的矩阵分解到最近出现的基于深度学习的方法，现有的努力通常通过从描述用户（或项目）的预先存在的特征（例如 ID 和属性）映射来获得用户（或项目）的嵌入。在这篇论文中将用户-项目交互——更具体地说是二分图结构——集成到嵌入过程中。这篇论文开发了一个新的推荐框架神经图协同过滤 (NGCF)，它通过在其上传播嵌入来利用用户-项目图结构。这导致了用户-项目图中高阶连接的表达建模，有效地将协作信号以显式方式注入嵌入过程。

算法的基本原理：NGCF 的模型如图5所示，它的传播过程分为 message construction（消息构造）和 message aggregation（消息聚合）两部分。按照图中的结构图可以进行分以下三层：1.Embeddings：对 user 和 item 的嵌入向量，用 id 来嵌入就可以了 2.Embedding Propagation Layers：挖掘高阶连通性关系来捕捉交互以细化 Embedding 的多个嵌入传播层 3.Prediction Layer：用更新之后带有交互信息的 user 和 item Embedding 来进行预测。

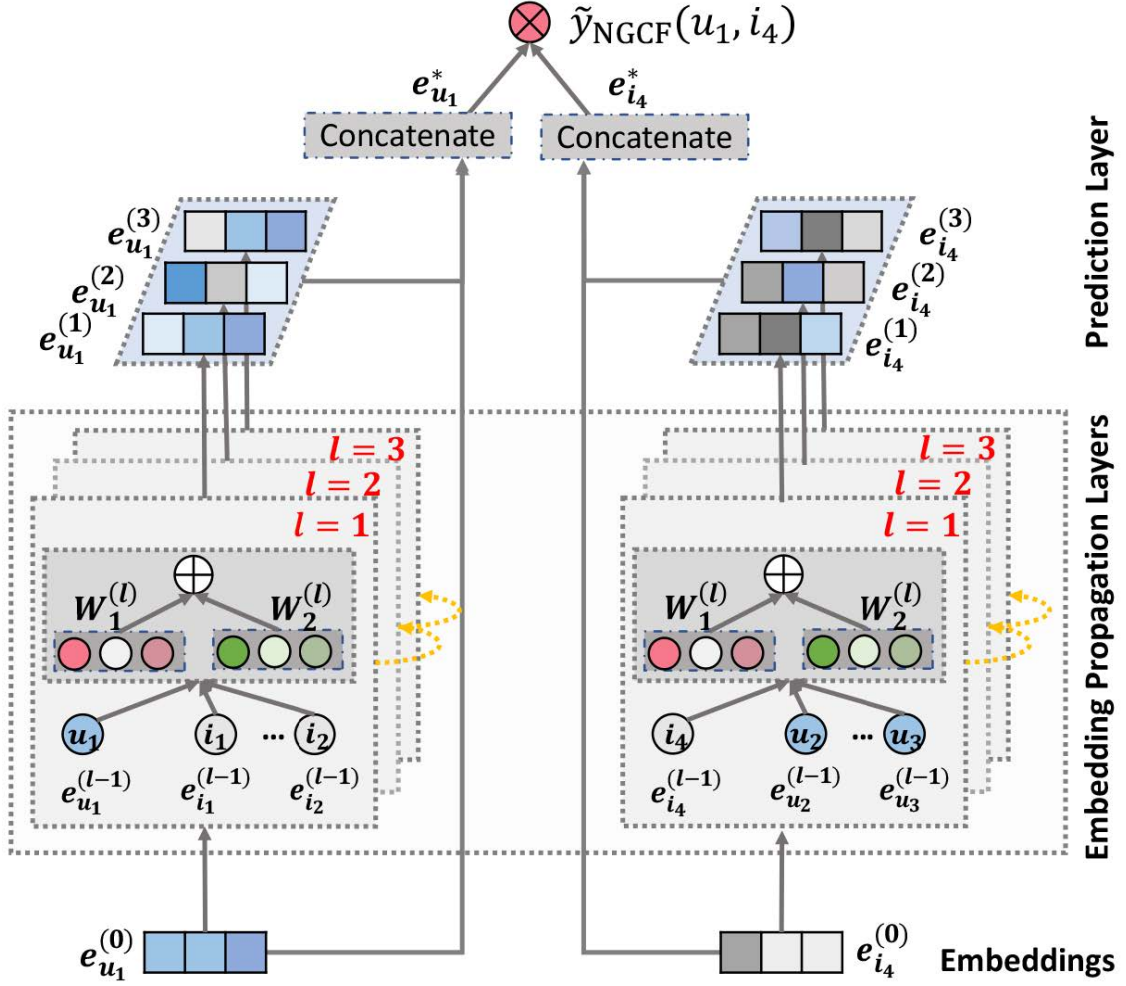


Figure 5: An illustration of NGCF model architecture.

3.2 LightGCN 算法原理和实现

在阅读了《Lightgcn: Simplifying and powering graph convolution network for recommendation》[2] 论文的基础上，整理了如下知识。

该论文旨在简化 GCN 的设计，使其更简洁，更适合推荐。该论文提出了一个名为 LightGCN 的新模型，仅包括 GCN 中最重要的组件——邻域聚合——用于协同过滤。具体来说，LightGCN 通过用户在用户-项目交互图上线性传播用户和项目嵌入来学习它们，并使用在所有层学习的嵌入的加权和作为最终嵌入。这种简单、线性 and 整洁的模型更容易实施和训练，与神经图协同过滤 (NGCF)——一种最先进的基于 GCN 的推荐系统相比，表现出实质性的改进（平均相对改进约 16.0%）模型——在完全相同的实验设置下。

和 NGCF 的不同点在于：lightGCN 将 GCN 中最常见的两种设计：特征转换和非线性激活弃用，因为他们对模型并无实质性作用，另外 LightGCN 认为自信息的作用不大，也没有使用自信息链接。整体模型示意图如图6所示。

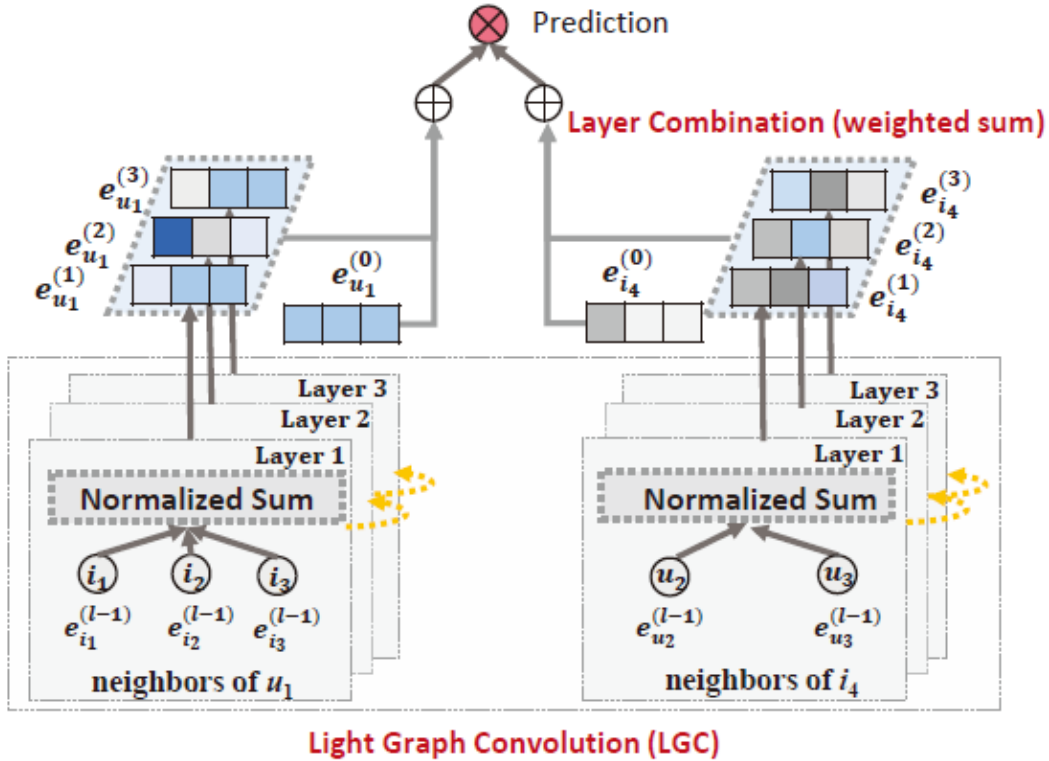


Figure 6: : An illustration of LightGCN model architecture

3.3 DHCF 算法原理和实现

在阅读了《Dual channel hypergraph collaborative filtering》[3] 论文的基础上，整理了如下知识。

论文提出了一个双通道超图协同过滤 (DHCF) 框架来解决上述问题。首先，双通道学习策略，引入整体利用分而治之策略的方法来学习用户和项目的表示，以便这两种类型的数据可以优雅地互连，同时仍保持其特定属性。其次，采用超图结构对具有显式混合高阶相关性的用户和项目进

地介绍了深度学习在推荐系统中的应用，并对未来的研究方向提出了展望。

文章主要涵盖以下内容：1. 引言：介绍推荐系统的背景和重要性，以及深度学习在推荐系统中的潜力。2. 深度学习在推荐系统中的应用：概述了深度学习方法在推荐系统中的各种应用，包括基于内容的推荐、协同过滤、个性化排序等。3. 深度学习模型：介绍了在推荐系统中常用的深度学习模型，如多层感知器（MLP）、卷积神经网络（CNN）、循环神经网络（RNN）、自编码器（Autoencoder）等，并讨论它们的优势和局限性。4. 深度学习推荐系统的挑战：探讨了深度学习推荐系统面临的一些挑战，如数据稀疏性、冷启动问题、可解释性等，并介绍了一些应对策略。5. 新的研究方向和展望：提出了一些新的研究方向和展望，如深度强化学习在推荐系统中的应用、多模态推荐、社交推荐等，并探讨了未来可能的发展趋势。

在本文中由于数据集是基于隐式反馈的，并且并没有用户之间的社交网络。所以我阅读了 ARP, CDAE 和 NeuMF 的实现原理。本部分接下来将对这三种模型进行详细的解释。

4.1 ARP 算法的原理和实现

在阅读了《Adversarial personalized ranking for recommendation》[4] 论文的基础上，整理了如下知识。

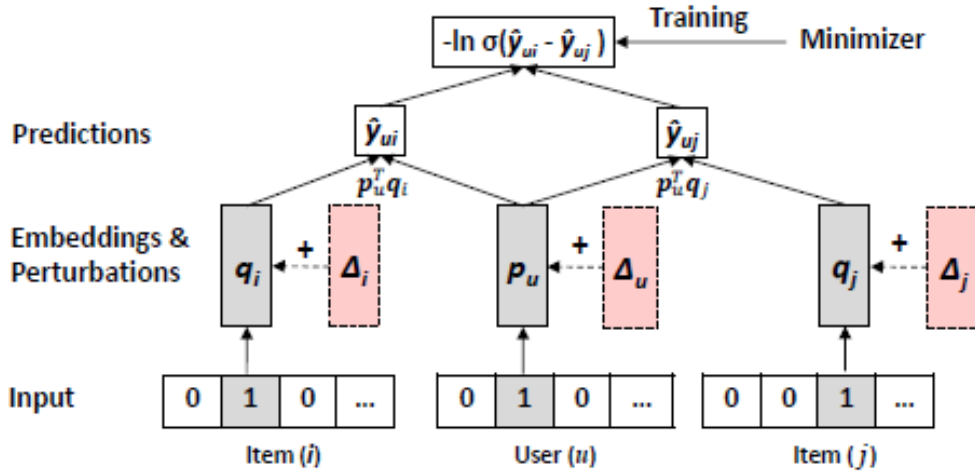


Figure 8: Illustration of our AMF method.

项目推荐是一项个性化的排名任务。为此，许多推荐系统优化具有成对排名目标的模型，例如贝叶斯个性化排名 (BPR)。但是使用 BPR 对其进行优化会导致推荐模型不稳健。特别是，该篇论文发现生成的模型极易受到其模型参数的对抗性扰动的影响，这意味着泛化可能存在较大误差。为了增强推荐模型的鲁棒性，从而提高其泛化性能，该篇论文提出了一种新的优化框架，即对抗性个性化排名 (APR)。简而言之，该论文的 APR 通过进行对抗训练来增强成对排序方法 BPR。它可以解释为玩一个 minimax 游戏，其中 BPR 目标函数的最小化同时防御对手，对手在模型参数上添加对抗性扰动以最大化 BPR 目标函数。为了说明它是如何工作的，我们通过在用户和项目的嵌入向量上添加对抗性扰动来在 MF 上实现 APR。在三个公共真实世界数据集上进行的大量实验证明了 APR 的有效性——通过使用 APR 优化 MF，它优于 BPR，平均相对提高 11.2%，并在项目推荐方面实现了最先进的性能。

模型中用到的对抗矩阵 (Adversarial Matrix Factorization) 示意图如图8所示。

4.2 CDAE 算法的原理和实现

在阅读了《Collaborative denoising auto-encoders for top-n recommender systems》[5] 论文的基础上，整理了如下知识。

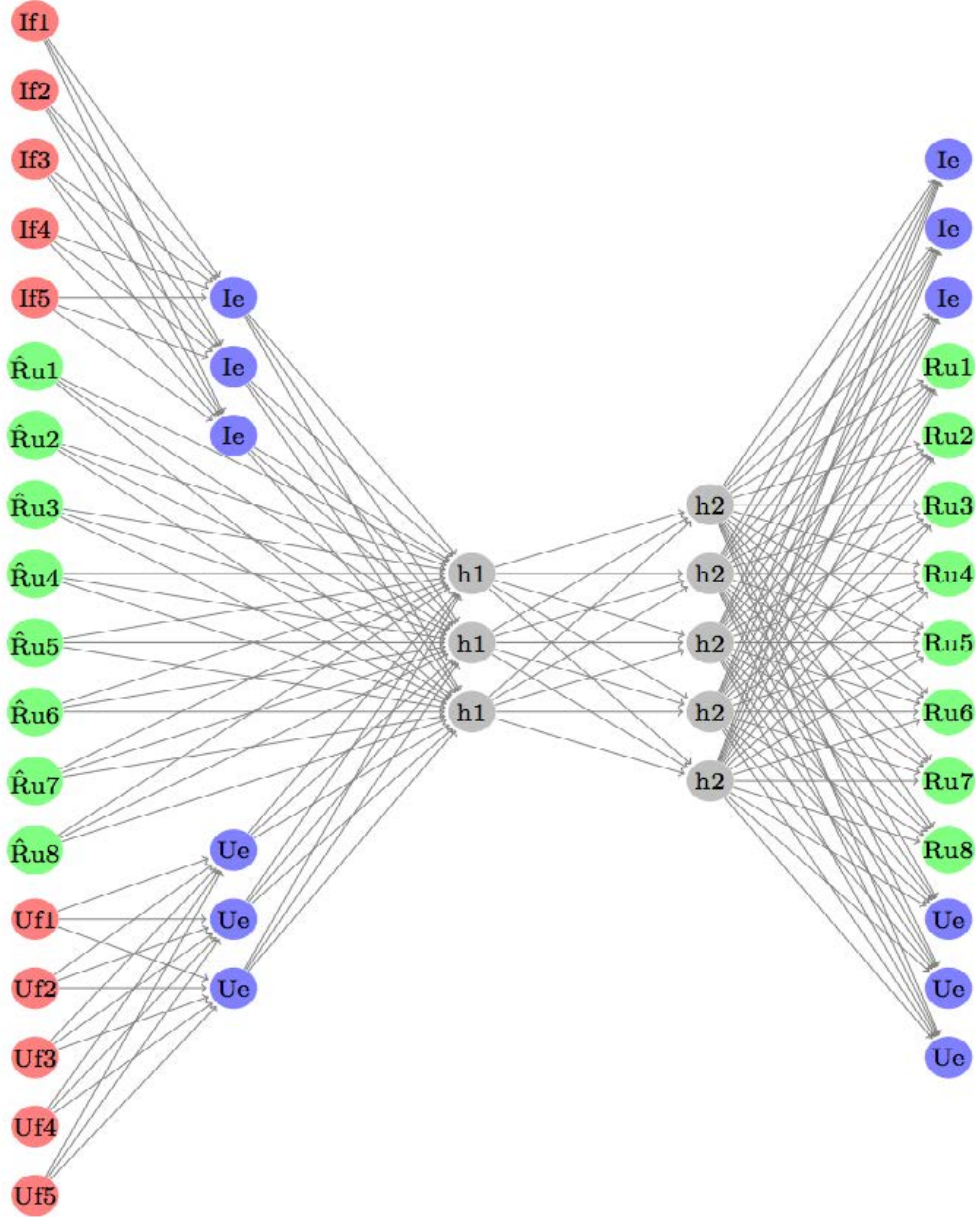


Figure 9: CDDAE framework.

该论文提出了一种称为协作去噪自动编码器 (CDAE) 的新方法，用于利用去噪自动编码器的思想进行 Top-N 推荐。该论文证明了所提出的模型是几个著名的协同过滤模型的概括，但具有更灵活的组件。进行了彻底的实验以了解 CDAE 在各种组件设置下的性能。

CDDAE 框架的示意图如图9所示。

4.3 NeuMF 算法的原理和实现

在阅读了《Neural collaborative filtering》[6] 论文的基础上，整理了如下知识。

该论文考虑到-虽然最近的一些工作已将深度学习用于推荐，但他们主要将其用于对辅助信息建模，例如项目的文本描述和音乐的声学特征。在对协同过滤中的关键因素——用户和物品特征的交互进行建模时，他们仍然采用矩阵分解，并对用户和物品的潜在特征应用内积。

通过用可以从数据中学习任意函数的神经架构替换内积，该论文提出了一个名为 NCF 的通用框架，NCF 是基于神经网络的协同过滤的缩写。NCF 是通用的，可以在其框架下表达和推广矩阵分解。为了增强非线性的 NCF 建模，该论文建议利用多层感知器来学习用户-项目交互函数。

NeuCF 框架的示意图如图10所示。

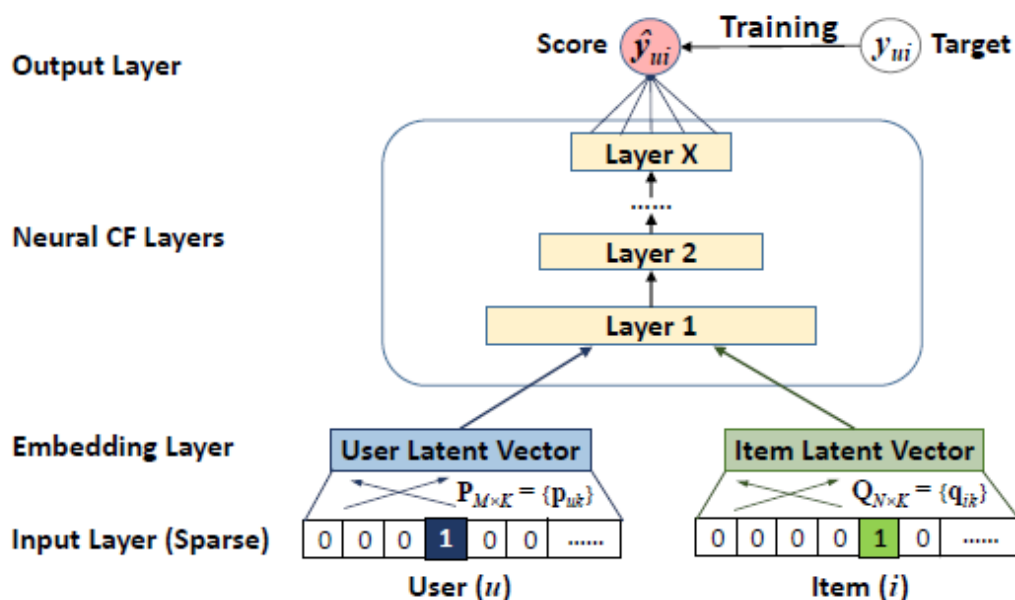


Figure 10: Neural collaborative filtering framework.

5 算法 python 实现、结果比对和最终的预测结果

5.1 算法的 python 实现

结合[GitHub 上开源项目 QRec](#)对六种算法模型进行 Python 代码实现。

- 在 config 目录下是算法模型需要参数的配置文件
- dataset 目录下是用到的数据集包括训练数据集和测试数据集
- dataset 目录下是用到的数据集包括训练数据集和测试数据集
- model 目录下是模型算法的实现代码
- results 目录下是实验结果
- util 目录下的 dataSplit.py 是我们划分训练数据集的代码

- 主目录下的 main.py 是程序入口、QRec.py 中是对模型算法的初始化

5.2 算法在验证集上的结果对比

我的实验思路是将老师给定的数据集进一步划分，取 80% 训练集数据为新的训练集和 20% 数据为验证集，然后以新的训练集训练六种推荐算法模型，并在验证集上验证推荐效果，最终选择在验证集上最优的推荐算法对测试集内用户进行 top-10 推荐预测。

划分数据集的代码在 util 包下的 dataSplit 方法下，将划分后的数据集输出到 genDatasets 目录下。

```
The result of NGCF [1]:  
Top 10  
Precision:0.09527472527472527  
Recall:0.13621059271667885  
F1:0.11212310925974986  
NDCG:0.1435686972347789
```

Figure 11: NGCF 算法模型的结果

```
The result of LightGCN [1]:  
Top 10  
Precision:0.12472527472527473  
Recall:0.17229795924017  
F1:0.14470188081875437  
NDCG:0.18564614360077014
```

Figure 12: LightGCN 算法模型的结果

```
The result of DHCF [1]:  
Top 10  
Precision:0.09681318681318682  
Recall:0.1356629226505956  
F1:0.11299191047621322  
NDCG:0.14239791995866172
```

Figure 13: DHCF 算法模型的结果


```
The result of APR [1]:  
Top 10  
Precision:0.08340659340659341  
Recall:0.11357716199802374  
F1:0.09618137446497191  
NDCG:0.11851551455414232
```

Figure 14: APR 算法模型的结果

```
The result of CDAE [1]:  
Top 10  
Precision:0.07725274725274725  
Recall:0.08687256131945771  
F1:0.08178073304103937  
NDCG:0.11660649499521036
```

Figure 15: CDAE 算法模型的结果

```
The result of NeuMF [1]:  
Top 10  
Precision:0.10637362637362638  
Recall:0.14932821810959016  
F1:0.12424301523778446  
NDCG:0.1609487198190751
```

Figure 16: NeuMF 算法模型的结果

实验指标采用 $Precision$, $Recall$, $F-measure$ 和 $NDCG$ 。实验过程中的结果截图如下, 其中图11是算法 NGCF 算法模型的结果, 图12是 LightGCN 算法模型的结果, 图13是 DHCF 算法模型的结果, 图14是 APR 算法模型的结果, 图15是 CDAE 算法模型的结果, 图16是 NeuMF 算法模型的结果:

整理的算法的性能对比见表1。从表1可以发现在验证集上表现的最好模型是 LightGCN 模型,

Table 1: 算法性能对比

评估指标\模型算法	NGCF	LightGCN	DHCF	APR	CDAE	NeuMF
Precision	0.095	0.125	0.097	0.083	0.077	0.106
Recall	0.136	0.172	0.136	0.114	0.087	0.149
F-measure(F1)	0.112	0.145	0.113	0.096	0.082	0.124
NDCG	0.144	0.186	0.142	0.119	0.117	0.161

其次是 NeuMF 模型, 另外四个模型的 Precision 指标都没有达到 0.1。所以我们选择 LightGCN 模型作为我们的模型预测测试集上用户的 Top-10 商品。

5.3 LightGCN 模型在测试集上的预测结果

因为 LightGCN 模型算法选取的训练参数如下:

```
num.factors=20
num.max.epoch=1000
batch_size=2000
learnRate=-init 0.001 -max 1
reg.lambda=-u 0.001 -i 0.001 -b 0.2 -s 0.2
```

参数的具体解释见下表2:

Table 2: LightGCN 参数的详细解释

参数	解释
num.factors	设置潜在因素的数量
num.max.epoch	设置迭代推荐算法的最大次数
batch_size	批处理大小
learnRate	-init initial 迭代推荐算法的初始学习率 -max: 最大学习率 (默认为 1);
reg.lambda	-u: 用户正则化;-i: 项目正则化;-b: 偏置正则化;-s: 社会正规化

设置 num.factors=20, batch_size=2000, learnRate 和 reg.lambda 均已设置, 现在就是调节 epoch 的大小。考虑将 num.max.epoch 设置为 600, 700 800, 1000, 1200, 查看在验证集上的效果, 以便选取最优的参数来对测试集上的用户做出预测。在验证集上的结果对比见下表3。所以我们最

Table 3: num.max.epoch 参数对算法性能的影响

评估指标\LightGCN(epoch=)	600	700	800	1000	1200
Precision	0.126	0.128	0.127	0.125	0.120
Recall	0.176	0.180	0.173	0.172	0.168
F-measure(F1)	0.150	0.124	0.147	0.145	0.140
NDCG	0.186	0.189	0.188	0.186	0.178

终选择 num.max.epoch=700 作为我们的参数进行预测用户的 Top-10 商品。实验结果放在了傅显坤 _result.txt 文件中。

通过这次课程实验我掌握了很多知识，之前只是听说过推荐系统，并没有深入的了解。通过课程学习学到了协同过滤等相关知识，并且了解了很多前沿技术。最主要的是通过实验帮助理解推荐系统理论，学习了六种算法模型的原理和实现，掌握了数据推荐过程中的数据处理、推荐算法以及算法评估指标等等。总之收获颇丰!!!

References

- [1] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*, pages 165–174, 2019.
- [2] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 639–648, 2020.
- [3] Shuyi Ji, Yifan Feng, Rongrong Ji, Xibin Zhao, Wanwan Tang, and Yue Gao. Dual channel hypergraph collaborative filtering. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2020–2029, 2020.
- [4] Xiangnan He, Zhankui He, Xiaoyu Du, and Tat-Seng Chua. Adversarial personalized ranking for recommendation. In *The 41st International ACM SIGIR conference on research & development in information retrieval*, pages 355–364, 2018.
- [5] Yao Wu, Christopher DuBois, Alice X Zheng, and Martin Ester. Collaborative denoising auto-encoders for top-n recommender systems. In *Proceedings of the ninth ACM international conference on web search and data mining*, pages 153–162, 2016.
- [6] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, pages 173–182, 2017.

- [7] Folasade Olubusola Isinkaye, Yetunde O Folajimi, and Bolande Adefowoke Ojokoh. Recommendation systems: Principles, methods and evaluation. *Egyptian informatics journal*, 16(3):261–273, 2015.
- [8] Shiwen Wu, Fei Sun, Wentao Zhang, Xu Xie, and Bin Cui. Graph neural networks in recommender systems: a survey. *ACM Computing Surveys*, 55(5):1–37, 2022.
- [9] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. Deep learning based recommender system: A survey and new perspectives. *ACM computing surveys (CSUR)*, 52(1):1–38, 2019.