

分 类 号 _____
学校代码 10487

学号 D201881076
密 级 _____

华中科技大学

博士学位论文

(学术型 专业型)

基于对比学习的推荐算法研究

学位申请人： 刘斌

学科专业： 信息与通信工程

指导教师： 王邦 教授

答辩日期： 202X 年 X 月 X 日

答辩委员会

	姓名	职称	单位
主席			
委员			

**A Dissertation Submitted in Partial Fulfillment of the Requirements
for the Degree of Doctor of Engineering**

**Contrastive Learning for Personalized
Recommendation**

Ph.D. Candidate : LIU BIN

Major : Information and Communication Engineering

Supervisor : Prof. WANG BANG

Huazhong University of Science and Technology

Wuhan 430074, P. R. China

October, 2023

独创性声明

本人声明所呈交的学位论文是我个人在导师的指导下进行的研究工作及取得的研究成果。尽我所知，除文中已标明引用的内容外，本论文不包含任何其他人或集体已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：

日期： 年 月 日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，即：学校有权保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权华中科技大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保 密 ，在 _____ 年解密后适用本授权书。

本论文属于

不保密 。

(请在以上方框内打“√”)

学位论文作者签名：

指导教师签名：

日期： 年 月 日

日期： 年 月 日

摘 要

在互联网飞速发展的信息时代，推荐系统在电商、社交媒体、新闻阅读等领域发挥着重要的作用，受到了广泛的关注和研究。推荐领域的排序预测任务根植于对比学习范式，通过编码不同样本之间的差异特征学习数据潜在的结构和特性，取得了出色的泛化能力。然而，推荐场景中的隐式反馈数据集具有正例弱监督、负例无监督等特性，对学习良好的用户、物品特征的表示产生了挑战，并对排序预测任务产生了不利的影响。如何从弱监督的用户行为数据中学到良好的用户、物品的特征表示，是对比学习基础理论的需要，也是提升推荐算法准确性的现实需求。

本文从推荐场景中正例弱监督、负例无监督的问题出发，针对对比学习的三个关键组件，正例、负例、损失函数做出了改进，提出了正例去噪算法、负例采样算法和损失函数校正算法，并且将损失函数校正方法从单个负例的成对学习向更具一般性的多个负例的对比学习推广。本文的创新和贡献如下：

针对伪正样本问题，提出了自适应学习的成对排序算法，用于从含噪成对比较的数据中学习个性化排序。将从含噪成对比较的数据中学习排序的问题形式化为一个含有隐变量的最大后验估计问题。所提出的算法采用期望最大化框架：在期望步骤中使用贝叶斯推断来估计交互的置信度指标；在最大化步骤中固定置信度指标，更新参数学习用户和物品的特征表示。该算法在合成的噪声数据集上表现出了更好的鲁棒性，在真实的数据集上表现出更高的推荐准确性。

针对伪负样本问题，提出了贝叶斯最优负采样算法，用于从未标记的数据中采样高质量的负例。提出了贝叶斯最优负采样准则，是最小化经验采样风险的理论最优的采样规则。该方法提出了后验概率意义上的负信号测度，结合了静态的先验信息和动态的样本信息，较好地综合了现有的两种提取负信号的两种主要范式，并为使用辅助信息建模先验概率的方法提供了灵活的接口。该算法在采样误差率和采样样本信息量以及推荐准确性上优于同类方法。

针对有偏的成对损失优化目标，提出了去偏成对学习算法，通过校正正例-未标注（Positive-unlabeled）数据计算得到的成对损失，以近似完全监督数据的成对损失函数，以获得接近于有监督学习的泛化性能。所提出的目标函数易于实现，不需要

华 中 科 技 大 学 博 士 学 位 论 文

额外的辅助信息进行监督，也不需要过多的存储和计算开销。在保持严格的相对于成对学习的线性时间复杂度情况下，去偏成对学习算法取得了更高的推荐准确性。

针对有偏的对比损失优化目标，提出了贝叶斯自监督对比学习算法，通过重要性权重来纠正从无标签数据中随机选择的负样本引入的偏差。从单个权重来看，该权重正比于未标注样本是真负例的后验概率估计，可以灵活统一地执行硬负例挖掘和伪负例去偏任务。从损失函数的数值来看，该方法是有监督对比损失的一致估计，从而提升下游任务的泛化性能。在数值实验、个性化推荐和图像分类等多个任务上验证了贝叶斯自监督对比学习算法的有效性。

综上，本文面向隐式反馈数据的弱监督特性，针对对比学习的三个关键组提出了相应的正例去噪算法、负例采样算法和损失函数校正算法，为后续推荐算法和对比学习研究提供了有效的支撑。

关键词：推荐算法；对比学习；成对学习；自监督学习；负采样

Abstract

In the rapidly evolving information age of the Internet, recommendation systems play a pivotal role in e-commerce, social media, news reading, and other sectors, progressively dominating the process of information discovery and garnering widespread attention and research. Ranking prediction tasks in the recommendation field are rooted in the contrastive learning paradigm, which learns the underlying structure and attributes of data by encoding differences between various samples, exhibiting impressive data efficiency and generalization capabilities. However, the scenario features of recommendations determine the weak supervision of positive examples and the unsupervised nature of negative examples in implicit feedback datasets. This poses a challenge to learning good user and item feature representations and adversely affects ranking prediction tasks. How to learn good user and item feature representations from weakly supervised user behavior data is a theoretical necessity for contrastive learning and a practical requirement for improving the accuracy of recommendation algorithms.

This paper starts from the weak supervision of positive examples and the unsupervised nature of negative examples in recommendation scenarios. It makes improvements for the three key components of contrastive learning: positive examples, negative examples, and loss function. Correspondingly, it proposes a positive example denoising algorithm, a negative example sampling algorithm, and a loss function correction algorithm. Moreover, it extends the loss function correction method from pairwise learning of a single negative example to more generalized contrastive learning with multiple negative examples. The innovations and contributions of this paper are as follows:

For the pseudo-positive sample problem, an adaptive learning pairwise ranking algorithm is proposed, used for learning personalized ranking from noisy pairwise comparison data. The problem of learning ranking from noisy pairwise comparison data is formalized into a maximum posterior estimation problem with hidden variables. The proposed algorithm adopts an expectation-maximization framework: using Bayesian inference to estimate

the confidence indicators of interactions in the expectation step, and updating parameters to learn the feature representations of users and items in the maximization step. This algorithm shows better robustness on synthetic noisy datasets and higher recommendation accuracy on real datasets.

For the pseudo-negative sample problem, a Bayesian Optimal Negative Sampling algorithm is proposed, used for sampling high-quality negative examples from unlabeled data. A Bayesian Optimal Negative Sampling criterion is proposed, which is the theoretically optimal sampling rule for minimizing empirical sampling risk. This method specifies a negative signal measure in the sense of posterior probability, combining static prior information and dynamic sample information, unifying the two main paradigms for extracting negative signals, and providing a flexible interface for modeling prior probabilities using auxiliary information. This algorithm outperforms similar methods in terms of sampling error rate, sampling sample information amount, and recommendation accuracy.

For the biased pairwise loss optimization objective, a debiased pairwise learning algorithm is proposed, which approximates the pairwise loss function of fully supervised data to achieve generalization performance close to supervised learning. The proposed objective function is easy to implement, does not require additional auxiliary information for supervision, and does not require excessive storage and computational overhead. While maintaining a strict linear time complexity relative to pairwise learning, the debiased pairwise learning algorithm achieves higher recommendation accuracy.

For the biased contrastive loss optimization objective, a Bayesian Self-Supervised Contrastive Learning algorithm is proposed, which corrects the bias introduced by randomly selected negative samples from unlabeled data through importance weighting. From the perspective of single weight, this method provides a posterior probability estimate that the sample is a true negative example, and can flexibly and uniformly perform the two conflicting tasks of hard negative example mining and pseudo-negative example debiasing. In terms of the value of the loss function, this method provides a consistent estimate of the contrastive loss under supervised data, thereby enhancing the generalization performance of downstream tasks. The effectiveness of the Bayesian Self-Supervised Contrastive Learning algorithm has been verified on multiple tasks, including numerical experiments, personal-

华 中 科 技 大 学 博 士 学 位 论 文

ized recommendation, and image classification.

In summary, this paper proposes corresponding positive example denoising algorithms, negative example sampling algorithms, and loss function correction algorithms for the three key components of contrastive learning, targeting the weak supervision characteristics of implicit feedback data. It provides effective support for subsequent recommendation algorithms and contrastive learning research.

Keywords: Recommendation, Contrastive Learning, Pairwise Learning, Self-supervised Learning, Negative Sampling

目 录

摘 要	I
Abstract	III
1 绪论	
1.1 研究背景与意义.....	(1)
1.2 国内外研究现状.....	(2)
1.3 存在的问题	(3)
1.4 研究内容	(5)
2 基础理论与相关工作	
2.1 自监督推荐算法（SSR）相关工作.....	(7)
2.2 对比学习相关工作	(10)
2.3 预备知识	(18)
3 正例去噪算法研究	
3.1 引言	(23)
3.2 自适应去噪的成对排序算法	(25)
3.3 算法实现与时间复杂度分析	(31)
3.4 收敛性理论分析	(32)
3.5 试验评估	(35)
3.6 本章小结	(39)
4 负例采样算法研究	
4.1 引言	(41)
4.2 负采样分析	(42)
4.3 贝叶斯最优负采样算法.....	(44)
4.4 算法实现与时间复杂度分析	(50)
4.5 试验评估	(51)
4.6 本章小结	(60)

华 中 科 技 大 学 博 士 学 位 论 文

5 成对损失函数校正研究

5.1	引言	(62)
5.2	成对损失优化目标分析	(63)
5.3	去偏成对学习算法	(65)
5.4	算法实现与时间复杂度分析	(67)
5.5	去偏成对损失的理论分析	(70)
5.6	实验评估	(75)
5.7	本章小结	(82)

6 对比损失函数校正研究

6.1	引言	(83)
6.2	对比损失优化目标分析	(84)
6.3	贝叶斯自监督对比学习算法	(87)
6.4	算法实现与时间复杂度分析	(97)
6.5	贝叶斯自监督对比损失的理论分析	(98)
6.6	实验评估	(104)
6.7	本章小结	(118)

7 总结与展望

7.1	全文总结	(120)
7.2	后续工作展望	(121)
致 谢		(123)
参考文献		(124)

附录 1	答辩委员会决议	(136)
------	---------	-------

附录 2	攻读博士学位期间取得的研究成果	(137)
------	-----------------	-------

附录 3	公开发表的学术论文与博士学位论文的关系	(138)
------	---------------------	-------

附录 4	攻读博士学位期间参与的科研项目	(139)
------	-----------------	-------

插图

1.1	关键问题与研究内容的关系	5
2.1	困难负例和伪负例示意图	12
2.2	次序统计量的概率微元	20
2.3	随机变量及其次序统计量分布示意图	21
3.1	含噪成对比较示意图	24
3.2	不同参数下先验分布示意图	29
3.3	不同参数下后验分布示意图	30
3.4	隐变量估计的 ROC 曲线	37
3.5	不同噪声率对学习算法性能的影响	38
3.6	不同学习算法的收敛速度比较	40
4.1	Positive-Unlabeled(PU) 问题示意图	41
4.2	不同训练时点的相似度分数的经验分布	45
4.3	相似度分数的理论分布	46
4.4	不同方法的采样质量比较	55
4.5	参数 λ 和 $ \mathcal{M}_u $ 的影响	57
5.1	偏倚的概率估计的示意图	65
5.2	去偏成对损失的示意图	67
5.3	不同超参数的影响	79
5.4	去偏成对学习算法和负采样算法的运行时间比较	81
6.1	有偏对比损失与互信息优化示意图	86
6.2	自监督对比学习的伪负例去偏和硬负例挖掘示意图	87
6.3	正负样本对相对位置示意图	90
6.4	不同 α 设定下类条件概率密度示意图	93
6.5	抽象数学模型示意图	99

华 中 科 技 大 学 博 士 学 位 论 文

6.6 估计校正的直观阐释示意图	103
6.7 未标注样本相似度分数生成过程示意图	105
6.8 不同 α 设置下相似度分数的经验分布	108
6.9 不同参数设置下估计量的均方误差比较	110
6.10 不同参数设置下估计量的均值比较	111
6.11 CIFAR10 数据集的十个类别及相应的随机样本概览	112

表格

3.1 数据集统计信息	35
3.2 隐变量的估计质量	36
3.3 Top-k 推荐性能比较	38
4.1 数据集统计信息	52
4.2 Top-k 推荐性能比较	54
4.3 BNS 负采样算法的消融实验	58
4.4 不同损失函数下 BNS 负采样算法的表现	59
4.5 向最优采样器的渐进过程	60
5.1 数据集统计	76
5.2 Top-k 推荐性能比较	78
5.3 去偏成对学习算法与负采样算法的性能比较	81
6.1 概率密度一览表	94
6.2 图像分类任务数据集统计信息	113
6.3 Top-k 推荐性能比较	115
6.4 Top-1 图像分类精度比较	116
6.5 负例个数的影响	116
6.6 参数 β 的影响	116
6.7 正例类先验概率 τ^+ 的影响	118
6.8 算法运行时间比较	118

中英文缩写对照表

英文缩写	英文全称	中文全称
SSL	Self-Supervised Learning	自监督学习
SL	Supervised Learning	监督学习
UL	UnSupervised Learning	无监督学习
PL	Pairwise Learning	成对学习
CL	Contrastive Learning	对比学习
PU	Positive-Unlabeled	正例未标注
HN	Hard Negative Sample	困难负例
FN	False Negative Sample	伪负例
TN	True Negative Sample	真负例
FP	False Positive Sample	伪正例
NS	Negative Sampling	负采样
NCE	Noisy Contrastive Estimation	噪声对比估计
PDF	Probability Density Function	概率密度函数
CCPD	Class Conditional Probability Density	类条件概率密度
CDF	Cumulative Distribution Function	累计分布函数
eCDF	Empirical Cumulative Distribution Function	经验分布函数
MLE	Maximum Likelihood Estimation	极大似然估计
MAP	Maximum Posteriori Estimation	最大后验估计
BPR	Bayesian Personalized Ranking	贝叶斯个性化排序

1 绪论

1.1 研究背景与意义

作为互联网信息时代解决信息过载问题的重要工具之一，推荐系统通过分析用户的交互数据和历史行为信息，可以提供相关性、多样性和新颖性的内容选择，能够帮助用户快速准确地找到感兴趣的信息，辅助用户做出更明智的决策。在互联网飞速发展的信息时代，推荐系统在电商、社交媒体、新闻阅读等领域发挥着重要的作用，逐步主导了人们的信息发现过程。因此，推荐算法具有重大研究意义和广阔的应用前景^[1-3]。

面向隐式反馈数据的学习排序 (Learning-to-Rank) 算法根植于对比学习的范式。其中，最广泛使用的成对学习是具有单个负例的对比学习的特例。在完全监督下，对比学习在推荐任务中有很大的优势，主要体现在以下几点：(1) 互信息优化：通过对语义上相似（正样本）和不相似（负样本）的样本对，拉近锚点与正例的距离，推远锚点与负例的距离，实现对变量互信息的优化。(2) 避免过拟合：对比损失激励编码器编码不同样本的差异特征，免于对样本进行像素级信息重构，从而有效的避免了过拟合。(3) 排序导向：通过优化成对损失或对比损失，实现了对排序指标 AUC 和 NDCG 的优化^①。此外，对比学习在计算机视觉、自然语言处理和信息检索领域取得了巨大的成功，在分类、回归、排序等任务表现出了出色的泛化性能。

受限于推荐系统的场景特征，获取准确的正负样本标签用于对比学习是不实际且昂贵的。传统的有监督学习在推荐中遇到了严重的瓶颈：它不仅严重依赖昂贵的手动标注，而且还面临泛化误差、虚假相关性和对抗性攻击等问题。目前许多最先进的模型都通过对输入数据进行某种变换或处理来创建“伪标签”。通用的做法是将某个用户已交互的物品被标记为正样本，未交互的物品被标记为负样本^[1,4-6]。类似地，在计算机视觉中，某个样本的增强被标记为正样本，其他样本被标记为负样本^[7-10]。

基于自监督技术创建的伪标签，一方面摆脱了对密集人工标注数据集的依赖，使得从海量无监督数据中训练模型成为可能。但另一方面，隐式反馈数据通常是正

^① 如不考虑正则化项，成对损失函数 BPR 与 AUC 指标定义相同，不同之处在于 BPR 把 AUC 指标中不可微的 0 – 1 损失 $\ell_{0-1}(\cdot)$ 替换成了可微的替代损失 $\log \sigma(\cdot)$ ^[11]。

例-未标注 (Positive-Unlabeled, PU) 的形式，且正例中包含噪声，如代购、误触等。根据隐式反馈数据交互的有无创建的正负样本标签，不仅导致了伪正例和伪负例的问题，还使得自监督对比学习设置下的优化目标偏离了原始的优化目标，对准确地学习样本的特征表示提出了严峻挑战。如何从弱监督的用户行为数据中学到良好的用户、物品特征表示，是对比学习基础理论的需要，也是提升个性化推荐系统准确性的现实需求。

1.2 国内外研究现状

对样本的度量方式主要有两种：一是对样本进行逐个度量，例如个体身高，体重，等。二是在很多高自由度场景中，由于：(1) 单个样本的绝对值度量不容易获取；(2) 有些度量体现相互关系，只能以成对方式体现；(3) 特定场景中更关注相对值的大小而非绝对值的大小，导致了另外一种主要的度量方式出现：成对度量 (pairwise measurement)。例如方案 i 比方案 j 好，用户购买了 i 但是没有购买 j 。成对度量广泛存在于描述客观世界的数据中，度量的方式主要有：(i) 成对比较，体现的是强弱关系；(ii) 成对关系，例如网页 a 的超链接指向网页 b ，体现了相互依存和作用的关系。成对度量提供了从相互关系描述、分析、挖掘事物的视角，相比于逐个度量的方式，突出了数据内部的结构信息，能够更好地描摹客观世界存在的真实形态。

由于用户很少显式地表达对物品的偏好，通常能收集到的数据是一种隐式反馈数据，如点击，购买，浏览等行为数据。贝叶斯个性化排序 (Bayesian Personalized Ranking, BPR)^[1] 率先将隐式反馈数据转化为成对比较 (Pairwise Comparision)。具体而言，对于某个用户（锚点），观测到的交互物品 i 被标记为正例，无交互的物品 j 被标记为负例，相应的正负例构成一个成对比较实例，意为用户对正例的偏好强于负例。基于构建的成对比较，BPR 使用 Bradley-Terry 模型^[11,12] 建模成对比较的似然，并通过极大似然估计学习潜在的用户表示和物品表示，从而以为每个用户预测个性化排序。BPR 通过将评分预测任务转化为排序预测任务，解决了交互数据无偏好值的问题。由于 BPR 直接针对排序目标 AUC 指标进行优化，逐渐主导了从隐式反馈数据中学习排序的任务^[13–18]。

这种从成对比较中学习排序的方法，在推荐中被称作成对学习。成对学习拥有广泛的应用领域^[11,19,20]，如决策制定^[21]、网络搜索^[22]、信息检索^[2]、偏好加总^[23]等。

值得指出的是，BPR 损失与噪声对比估计^[24,25]（Noisy Contrastive Estimation, NCE）损失具有完全等价的函数形式^[26]，它们都是广泛使用的对比损失 InfoNCE^[7] 负例个数为 1 的特例。尽管它们的解释不同，BPR 将优化目标解释为用户偏好正例强于负例的后验，InfoNCE 将优化目标解释为区分出正样本的后验，但实际上都描述了由模型参数化的正例得分大于负例得分的程度。随后，InfoNCE 损失也广泛应用于推荐中的排序预测任务^[27,28]。文献^[27]进一步讨论了优化 InfoNCE 损失导致优化排序指标 NDCG。

对比学习采用从比较中学习（learn-to-compare）^[24] 的范式，通过区分观察数据和噪声数据来使模型免于重建数据的像素级信息^[7]。虽然表示编码器和相似性度量会因任务而异^[5,6,9,29–31]，但它们都共享将正样本和负样本进行对比的基本思想，通过优化对比损失^[32] 来训练编码器，如 NCE 损失^[24]，InfoNCE 损失^[7]，Infomax 损失^[33]，渐进对比损失^[32] 等。这些损失函数隐式^[7] 或显式^[33] 地优化了已知数据和待预测数据互信息的下界。Arora 等人^[34] 为分类任务的对比学习的泛化界限提供了理论分析。在不同领域的许多应用中都观察到了监督对比学习的显著成功^[26,35–41]。

个性化推荐的排序预测根植于对比学习的范式，从对比学习的视角，排序预测任务后续的研究按照对比学习的关键组件可以归纳为，包括正例增强方法，负例采样方法，对比损失函数设计等工作。正例增强围绕着获取语义不变的正例用于对比的目标展开。通过对原图和增强图，学习体现用户物品二分图中重要的结构信息的节点特征表示^[28,42–44]；负例的采样方法围绕采样困难负样本^[45–49]，且避免采样伪负样本^[3,6,13,14,50–53] 这两个目标展开，以激励模型更准确地学习用户的兴趣边界。在损失函数设计上，主要是将对比损失进行线性组合，将原始的单一对比目标向多个对比目标进行拓展，使得最终的优化目标同时约束排序目标和保留图结构信息的目标^[28,42,43,54–57]。

1.3 存在的问题

在个性化推荐中，正例的语义为“用户喜欢的物品”，负例的语义为“用户不喜欢的物品”。然而，用户通常只通过交互行为（如点击、购买）来表达他们的偏好或兴趣。一方面，这些交互行为通常不包含显式的评分，只能观测到交互的有无，无法观测到交互数据所对应的偏好强度高低。大量不体现用户偏好的交互如代

华 中 科 技 大 学 博 士 学 位 论 文

购、误触、查看等交互数据与正常交互无法区分，都被标记为正例，称为正例的弱监督。另一方面，由于用户仅仅提供正反馈，无法观测到用户不喜欢哪些物品，所有未交互的物品连同用户潜在喜欢的物品都被标记为负例，但它们实际上是未标注数据，称为负例的无监督。因此，用于训练推荐系统的数据集通常以正例-未标注(Positive-Unlabeled, PU)形式存在，且正例还含有噪声。这些不完整标注和不准确标注的隐式反馈数据，对排序预测任务产生了不利影响，主要体现在以下方面：

- **正例弱监督：**不体现用户偏好的交互如代购、误触、查看等交互数据构成的伪正例可能导致模型对数据的过拟合。模型可能会将噪声视为真实模式，从而导致错误的学习和预测。更具体地，模型将用户不喜欢的物品误判为用户喜欢的物品，协同过滤机制会为用户推荐与伪正例类似的物品，从而导致 top-k 推荐列表中过高的伪正例率。
- **负例无监督：**由于负例的未标注，实际使用的负例来自未标注样本的数据分布，与负例的分布存在差异，模型无法准确地捕捉负例的真实分布和模式，从而导致预测结果的不可靠性。作为负例使用的未标注样本包含大量用户没看到但是潜在感兴趣的物品，这些伪负例会导致模型误判用户不喜欢此类物品，并减少对此类物品的推荐，从而导致 top-k 推荐列表过低的真正例率。
- **成对学习优化目标偏离：**在完全监督下，成对学习的原始优化目标是排序列表 AUC 风险的无偏估计，最小化成对损失将导致 AUC 最大化。在负例无监督情况下，使用未标注样本计算的成对损失优化目标偏离了原始优化目标，不再是 AUC 风险的无偏估计。
- **对比学习优化目标偏离：**在完全监督下，对比学习的原始优化目标含义是从负例中分类出正例的交叉熵。在负例无监督情况下，任何正例以外的样本都被视作负类，此时类的语义与传统的类的语义发生偏离，导致实际的优化目标是从未标注样本中区分出正例的交叉熵。

上述三个问题共同导致推荐模型误判用户的兴趣边界，学到不准确的用户、物品特征表示，从而产生不准确的排序预测，对推荐系统的准确性造成不利影响。

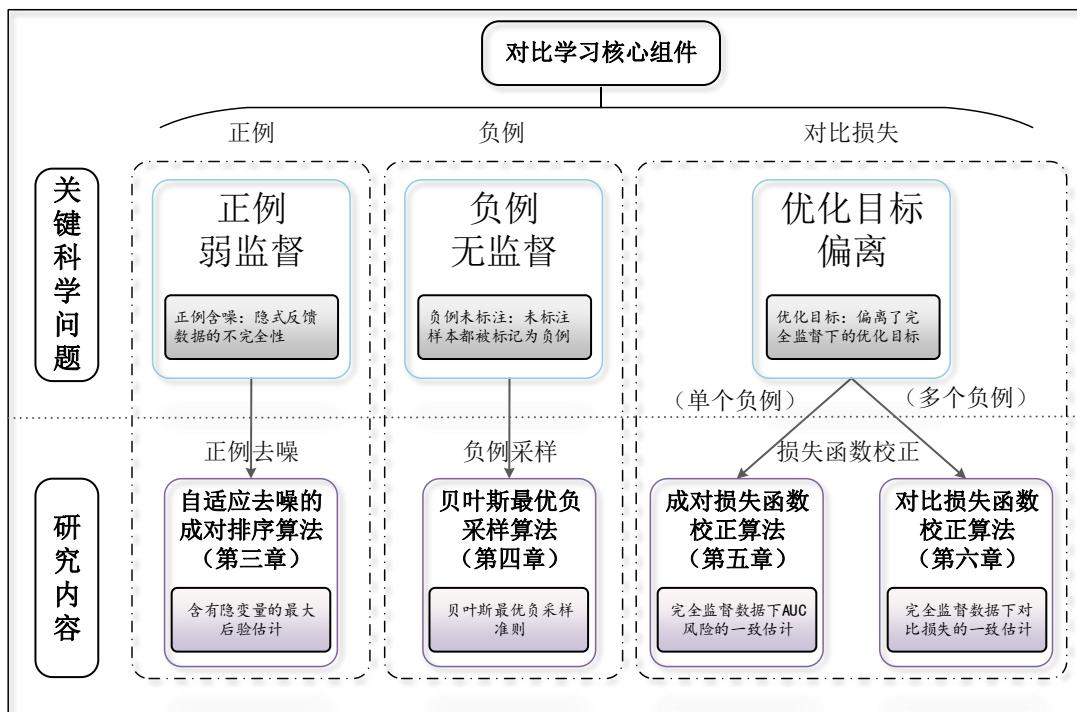


图 1.1 关键问题与研究内容的关系

1.4 研究内容

本文从推荐场景中正例弱监督、负例无监督的问题出发，针对对比学习的三个关键组件正例、负例、损失函数进行了研究，提出了正例去噪算法、负例采样算法和损失函数校正算法，并且将损失函数校正方法从只有一个负例的成对学习向更具一般性的多个负例的对比学习推广。图1.1展示了本文的研究内容和关键问题的关系。本文的创新和贡献如下：

针对伪正样本问题，提出了自适应学习的成对排序算法，用于从含噪成对比较的数据中学习个性化排序。分析了隐式反馈的本质是数据不完全 (incomplete data)，并把从含噪成对比较的数据中学习排序的问题形式化为一个含有隐变量的最大后验估计问题。引入一个隐变量作为衡量交互置信度的新指标，这个隐变量与用户物品的特征表示一起作为模型参数一起端到端地学习。所提出的算法采用期望最大化框架：在期望步骤中使用贝叶斯推断来估计交互的置信度指标；在最大化步骤中固定置信度指标，更新参数学习用户和物品的特征表示。该算法在合成的噪声数据集上

华 中 科 技 大 学 博 士 学 位 论 文

表现出了更好的鲁棒性，在真实的数据集上表现出更高的推荐准确性。

针对伪负样本问题，提出了贝叶斯最优负采样算法，用于从未标记的数据中采样高质量的负例，以提升对比学习的训练效果并提升推荐精度。提出了贝叶斯最优负采样准则，这是最小化经验采样风险的理论最优的采样规则。该方法指定了后验概率意义上的负信号测度，结合了静态的先验信息和动态的样本信息，统一了现有的两种提取负信号的两种主要范式，并为使用辅助信息建模先验概率的方法提供了灵活的接口。该算法在采样误差率和采样样本信息量以及推荐准确性上优于同类方法。

针对有偏的成对损失优化目标，提出了去偏成对学习算法，以近似完全监督数据的成对损失函数，以获得接近于有监督学习的泛化性能。该方法通过校正伪负样本导致的概率估计偏差，从而修正梯度以近似完全监督数据的梯度。所提出的目标函数易于实现，不需要额外的辅助信息进行监督，也不需要过多的存储和计算开销。在保持严格的相对于成对学习的线性时间复杂度情况下，去偏成对学习算法取得了更高的推荐准确性。

针对有偏的对比损失优化目标，提出了贝叶斯自监督对比学习算法，通过重要性权重来纠正从无标签数据中随机选择的负样本引入的偏差。从单个权重来看，该方法提供了样本是真负例的后验概率估计，可以灵活统一地执行硬负例挖掘和伪负例去偏任务这两个冲突任务。从损失函数的数值来看，该方法提供了有监督数据下对比损失的一致估计，从而提升下游任务的泛化性能。在数值实验、个性化推荐和图像分类等多个任务上验证了贝叶斯自监督对比学习算法的有效性。只需要简单地修改损失函数，而无需额外的存储和计算开销。

2 基础理论与相关工作

不完全标注的隐式反馈数据是制约推荐模型充分发挥潜力的瓶颈。自监督学习 (SSL) [9] 作为一种学习范式，可以减少对手动标签的依赖，并能够在海量无标签数据上进行训练。推荐系统中最先进的模型也借鉴了自监督学习的技术，使得从稀疏的隐式反馈数据集上训练推荐系统成为可能。本章首先从自监督学习的角度对推荐系统的现有工作进行梳理。其次对自监督学习的主要途径—对比学习基础理论与相关工作进行介绍。最后，介绍本文会使用到的一些重要的预备知识。

2.1 自监督推荐算法 (SSR) 相关工作

近年来，借助深度神经网络模型的强大拟合能力和泛化性能，推荐系统取得了巨大的成功。然而，基于深度神经网络模型的推荐模型需要大量的训练数据。与可以通过众包完成的图像注释任务不同，推荐系统中的数据获取成本很高，因为个性化推荐依赖用户自身生成的数据，需要准确标注用户对海量物品的喜欢/不喜欢程度，才能使得推荐模型准确捕捉用户的偏好，从而产生准确的推荐。不幸的是，大多数用户通常只与众多物品中的一小部分进行交互 [7]。因此，隐式反馈数据的稀疏性和标注不完全性问题成为制约深度推荐模型充分发挥潜力的瓶颈 [8]。

自监督学习 (SSL) [9] 作为一种学习范式，充分利用输入数据中的内在关系，通过对输入数据进行某种变换或处理来创建“伪标签”，可以减少对手动标签的依赖，最近受到了相当大的关注。SSL 的基本思想是通过精心设计的自监督任务（即预训练任务），从丰富的无标签数据中提取可迁移的知识，其中监督信号是半自动生成的。由于能够克服广泛存在的标签不足问题，SSL 已经应用于包括视觉表示学习 [10], [11], [12]，语言模型预训练 [13], [14]，音频表示学习 [15]，节点/图分类 [16], [17] 等各个领域。特别是对比学习 (CL) 的复兴 [28] 显着推动了 SSL 的前沿。最新的基于自监督学习的方法甚至在许多计算机视觉 (CV) 和自然语言处理 (NLP) 任务中表现出与有监督模型相当的性能 [11], [27]。由于自监督学习的原则与推荐系统对更多注释数据的需求相吻合，受到自监督学习在其他领域取得的巨大成功的启发，目前已经有大量不断增长的研究探索将 SSL 应用于推荐领域。

华 中 科 技 大 学 博 士 学 位 论 文

自监督推荐（Self-Supervised Recommendation, SSR）模型为克服推荐系统中的数据稀疏问题提供了一种新的途径。综合文献^[26] 和文献【ref】的归纳，总结了自监督推荐（SSR）的三个关键特征如下：

- 半自动地利用原始数据本身进行标注。自监督推荐充分利用输入数据中的内在关系，通过对输入数据进行某种变换或处理来创建“伪标签”。例如 BPR^[1] 将交互的物品自动标记为正例，而未交互的物品标记为负例，就是在执行自动化标注过程。需要注意的是，这个标签是伪标签，不同于用户喜欢与否的真实偏好标签。
- 通过提取自监督信号对推荐模型进行（预）训练的任务。常见的自监督信号如 BPR 中“用户对正例的偏好强于负例”，常用用于优化成对损失。另一种常见的自监督信号是用户物品二分图的图数据增强和原图的语义不变性，常用于优化对比损失优化目标。
- 自监督任务旨在提升推荐性能，而非作为最终目标。这一特征强调了推荐任务和自监督任务之间的主辅关系，学习良好的用户物品特征表示，是为了产生更准确的个性化推荐结果。

根据其自监督任务的性质，可以将现有的自监督推荐模型分为四个类别：对比型、预测型、生成型和混合型。下文对自监督模型的类别进行介绍。

2.1.1 对比型推荐算法

根据自监督信号的来源，对比型的推荐算法可以分为三种类型：结构层对比、特征层对比和模型层对比。

2.1.1.1 结构层对比

结构层对比的自监督信号主要来自于图结构扰动的语义不变性，核心思想是：对图数据（Graph）进行轻微扰动，可能会导致类似的语义。通过对不同的图数据，可以获得对结构扰动的共享不变性作为自监督信号。本文遵循文献 [35], [36] 提出的分类法，将结构层对比分为两类：同尺度对比和跨尺度对比。同尺度对比涉及同一尺度上两个对象的视图，并进一步分为两个级别：局部-局部对比（Local-Local Contrast），全局-全局对比（Global-Global Contrast）。跨尺度对比涉及来自不同尺度上两个对象的视图，并进一步分为局部-全局对比（Local-Global Contrast）和局部-上

华 中 科 技 大 学 博 士 学 位 论 文

下文对比 (Local-Context Contrast)。

对于局部-局部对比 (Local-Local Contrast)，基于 dropout 的增强方法是创建扰动局部视图的最常用方法。作为代表性模型，SGL [31] 将节点 dropout、边 dropout 和随机游走增强应用于用户-物品二分图。它使用相同类型的增强操作生成两个增强图，并使用共享的图 LightGCN[84] 编码器学习节点嵌入。节点级对比使用 InfoNCE 损失 [15] 进行优化，采用批内负采样，并与推荐的贝叶斯个性化排名 (BPR) 损失 [85] 进行联合优化。类似地，DCL [86] 使用随机边 dropout 来扰动节点的 L-hop 邻居网络，生成两个增强的邻域子图。然后，它最大化在这两个子图上学习到的节点表示之间的一致性。HHGR [62] 提出了一种双尺度节点 dropout 方法，用于群组推荐 [87]。该方法包括粗粒度和细粒度的 dropout 方案，分别从所有群组中删除一部分用户节点，以及仅从特定群组中随机选择成员节点。然后，它最大化从这两个具有不同 dropout 粒度的视图学习到的用户节点表示之间的互信息。此外，KGCL [88] 将 dropout 应用于知识图，并提出了一种知识感知对比方法，对比从用户-物品图和知识图的增强中学习到的节点表示。

对于全局-全局对比 (Global-Global Contrast)，通常在序列推荐模型中使用，其中序列被视为用户的全局视图。作为代表性模型，CL4SRec [59] 使用三种随机增强方法——物品屏蔽、物品裁剪和物品重新排序——来获取序列的增强视图。CoSeRec[61] 用相关项目替换短序列中的项目，或将相关项目插入短序列中进行鲁棒的数据增强，而 ContraRec[92] 不仅对比从相同输入增强的序列，还将具有相同目标项目的序列视为正样本。此外，DHCN [29] 通过建模会话内和会话间的结构信息，创建给定会话的两个基于超图的视图。

对于局部-全局对比 (Local-Global Contrast)，旨在将全局信息编码到局部结构表示中，并统一全局和局部语义。它经常在图学习场景中使用。EGLN [67] 提出通过将合并的用户-物品对表示与全局表示进行对比来实现局部-全局一致性，其中全局表示是所有用户-物品对表示的平均值。类似地，BiGI [69] 执行局部-全局对比，但仅对其 h-hop 子图进行特征聚合以生成用户-物品对表示。在 HGCL [93] 中，构建了用户和物品节点类型特定的同质图。对于每个同质图，它使用 DGII [17] 过程最大化图的局部补丁与整个图的全局表示之间的互信息。它还提出了跨类型对比来衡量不同类型同质图之间的局部和全局信息。

对于局部-上下文对比 (Local-Context Contrast)，旨在将全局信息编码到局部信

息的上下文表示中，并统一局部和上下文的语义。受 [79] 的启发，NCL [78] 设计了一种原型对比目标，以捕捉用户/物品与其原型之间的相关性，原型表示一组语义邻居。原型是通过使用 K-means 算法对所有用户或物品嵌入进行聚类获得的，并使用 EM 算法递归调整原型。这一思想也体现在序列推荐中 [77]，其中语义原型被建模为用户意图，而所属序列是原型的局部视图。S3-Rec [26] 应用项目屏蔽和项目裁剪来增强序列，并设计了四个对比任务来预训练双向 Transformer 进行下一个项目的预测：项目-属性互信息最大化（MIM）、序列-项目 MIM、序列-属性 MIM 和序列-序列 MIM。

2.2 对比学习相关工作

2.2.1 对比学习基础概念

从统计学习的角度来看，机器学习模型可以分为两类：生成模型和判别模型^[58]。给定特征 X 和标签 Y 的联合分布 $P(X, Y)$ ，生成模型旨在建模

$$p(X|Y = y) = \frac{p(X, Y)}{p(Y = y)}$$

而判别式模型旨在建模

$$p(Y|X = x) = \frac{p(X, Y)}{p(X = x)}$$

生成式模型需要重构特征 X 的像素级信息，从而捕捉特征之间的潜在关系。而判别式则是直接对后验概率建模，旨在激励编码器编码不同类别样本之间的差异特征，不编码像素级特征。对比学习属于判别式的一种，旨在通过噪声对比估计（Noise Contrastive Estimation, NCE）^[24] 的目标函数在比较中学习（learn to compare）

$$\mathcal{L}_{\text{NCE}} = \mathbb{E}[-\log \frac{\exp(f(x)^T f(x^+))}{\exp(f(x)^T f(x^+)) + \exp(f(x)^T f(x^-))}] \quad (2.1)$$

其中， x 是锚点，在推荐中通常选择用户 u ， x^+ 是锚点的正例，在推荐中通常选择为已交互的物品 i ，于是 (u, i) 构成正例对； x^- 为锚点的负例，在推荐中通常选择为未交互的物品 j ，于是 (u, j) 构成负例对。 f 是编码器， $f(x)^T f(x^+)$ 是由编码器参数化的正例对相似度得分，通常为内积相似度或者余弦相似度，记为 \hat{x}_{ui} 。类似地， $f(x)^T f(x^-)$ 是负例对似度得分，记为 \hat{x}_{uj} 。可以看到，NCE 损失函数与 BPR 损失^[1]

的联系：

$$\begin{aligned}
 \mathcal{L}_{\text{NCE}} &= \mathbb{E}[-\log \frac{\exp(\hat{x}_{ui})}{\exp(\hat{x}_{ui}) + \exp(\hat{x}_{uj})}] \\
 &= \mathbb{E}[-\log \frac{1}{1 + \exp(\hat{x}_{uj} - \hat{x}_{ui})}] \\
 &= \mathbb{E}[-\log \sigma(\hat{x}_{ui} - \hat{x}_{uj})] \\
 &= \mathcal{L}_{\text{BPR}}
 \end{aligned} \tag{2.2}$$

其中 $\sigma(\cdot)$ 为 sigmoid 函数。在实践中，由于要引入正则化项避免过拟合，而正则化项又等价于高斯分布先验密度的对数，从而 BPR 将上式解释为观测到的有序对的最大后验估计。尽管 BPR 和 NCE 有着不同的解释，且编码器 f 的网络架构在不同任务中差异很大，但是 BPR 与 NCE 有着完全相同的数学形式，共享相同的优化目标。

需要说明的是，正例对或者负例对的概念在不同场景下差异很大，例如 SimCLR^[8] 将同一个图像的两次增强构造为正例对，BPR^[1] 将用户和交互过的物品构建为正例对，LightGCL^[28] 将增强图和原图中的同一个用户构造为正例对。具体的正例或负例语义可以根据任务以及场景进行迁移和拓展。将 NCE 中加入更多的负例，那么可以得到 InfoNCE 损失^[7]：

$$\mathcal{L}_{\text{INFONCE}} = \mathbb{E}[-\log \frac{\exp(f(u)^T f(i))}{\exp(f(u)^T f(i)) + \sum_{n=1}^N \exp(f(u)^T f(j_n))}] \tag{2.3}$$

可以看到，NCE 和 BPR 是 InfoNCE 负例个数 $N = 1$ 的特例。以最具一般性的 InfoNCE 损失为例，它的含义是正确分类正样本的交叉熵^[7]，最小的 InfoNCE 损失为 0，在正例相似度得分与负例相似度得分满足 $\exp(f(x)^T f(x_j^-) - f(x)^T f(x^+)) \rightarrow -\infty, j \in \{1, 2, \dots, N\}$ 时取到。这一结论也适用于 NCE 和 BPR，只是负例个数 $N = 1$ 。

进一步地，相似度分数与欧氏距离是一一对应的负相关关系。对于投影到单位超球面的两个 d 维向量 $\mathbf{x} = (x_1, x_2, \dots, x_d)$ 和 $\mathbf{y} = (y_1, y_2, \dots, y_d)$ 欧式距离 $d(\mathbf{x}, \mathbf{y})$ 与内积相似度分数 $\mathbf{x}^T \mathbf{y}$ 存在如下关系：

$$\begin{aligned}
 d(\mathbf{x}, \mathbf{y}) &= \sqrt{(x_1 - y_1)^2 + \dots + (x_d - y_d)^2} \\
 &= \sqrt{2 - 2\mathbf{x}^T \mathbf{y}}
 \end{aligned}$$

因此相似度分数越高，距离越近。

基于上述分析，可以从三个方面归纳对比学习的特征：

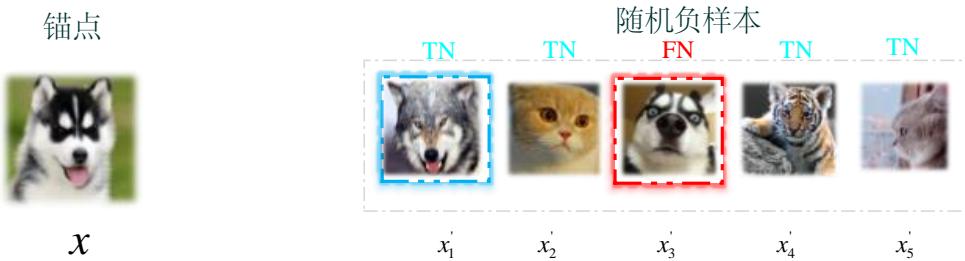


图 2.1 困难负例和伪负例示意图

- 从数值角度看，对比学习优化正例得分大于负例得分，只优化相似度分数的差值，不优化具体值，以激励学习数据的内在结构和特性。
- 从嵌入空间中看，对比损失激励编码器拉近正例与锚点的距离，推远负例与锚点的距离，实现编码不同样本的差异特征，而非样本本身的像素级信息。文献^[54] 和文献^[32] 分别阐释了优化 BPR 损失以及 InfoNCE 损失都是在优化正样本对齐、负样本的均匀。
- 从统计角度看，对比学习优化“正确分类正样本的概率”^[7]，体现判别式方法的特征，即直接对样本所属类别的后验概率建模。

2.2.2 负样本采样方法相关工作

对比学习有两个关键属性：正样本特征的对齐和负样本的均匀^[32]。如果仅仅优化正样本之间的距离，会使得任意样本之间的距离接近于零，即为所有的样本输出相同的表示，从而使得神经网络学到坍缩的解。由于大多数情况下缺乏明确的负信号，尤其是推荐系统中用户只对感兴趣的物品提供交互，因此负采样策略被广泛研究。以图2.1的一个直观的示例解释负采样策略的两个关键目标：锚点为狗，负例候选样本 x'_1 是狼，与锚点不同类，但是特征很相似，也称为困难负样本 (hard negative sample)。负例候选样本 x'_3 是狗，与锚点同类，但是由于缺乏监督标签也出现在负例候选集，称为伪负样本 (false negative sample)。为了使得同类样本的表示距离较近，那么 x'_1 狼这个困难负例应该被选作负样本，使其被推远，防止其混入狗这个类别；且 x'_3 狗这个伪负例不应该被选作负样本，防止其被推远。推荐领域的任务也是一样，只是正负样本语义是用户喜欢与否。负采样 (NS) 的策略是从候选的未标注样本中采样负样本，核心目标围绕如何采样困难负样本且剔除伪负样本展开。许多研究表明

华中科技大学博士学位论文

明，负采样对于提高下游的分类或推荐性能非常重要^[3,45,46,59–62]。

2.2.2.1 采样困难负样本

采样困难负样本的主要目标是采样类似于样本 x'_1 这样的困难负样本。它们与正例很相似，但是所属类别为负类。由于这两个样本在特征上相似度很高，难以区分，在经过网络提取特征之后，它们特征之间的相似度分数也比较高。采样此类样本一方面会使得不同类别的困难负样本被推远，迫使神经网络学习难以区分的样本之间的边界；另一方面，这类样本分数较高，梯度值较大，从而神经网络的参数更新更多，提高训练效率。基于上述动机，很多困难负采样 (hard negative sampling) 方法被提出，它们采用自适应采样分布的“困难负采样”，以针对困难负例进行采样^[3,45,46,59–61]。

在推荐领域，最具代表性的困难负采样方法为动态负采样^[46]。由于困难负样本的表示与正例比较接近，因此困难负样本对应的相似度分数也比较高。动态负采样通过在若干个候选的负例中采样相似度分数比较高的样本，从而实现每个样本的采样概率分布正比于其相对排序位置。类似地，Steffen 等人^[45] 提出排名位置靠前的负例进行过采样。他们的思想是类似的，因为排序位置与相似度分数是一一对应的。除了使用分数相似度或者相对排序位置来表征负例的困难等级 (hardness level)，还有研究者使用图的相关信息来采样困难负样本。例如，Wang 等人^[63] 和 Wang 等人^[64] 提出利用知识图谱上的关系类型来采样困难负例。另一种方使用较多的是选择与正例在图结构上相似的困难负例^[65–67]。文献^[49] 分析了，困难负采样实际上是在优化推荐列表指定伪正例例率 (FPR) 范围内的 AUC 面积 (one-way partial AUC)，但没有考虑困难负采样带来的伪负例问题。

2.2.2.2 剔除伪负样本

剔除伪负样本的主要目标是避免采样类似于样本 x'_3 这样的伪负样本。由于缺乏标签，它们虽然出现在负例候选集，但是所属类别为正类。在推荐领域，这类样本对应于用户没有看到过，但是潜在喜欢的物品。由于这个样本所属类别为正类，因此应当避免采样到此类样本，防止其在嵌入空间被推远。另一方面，采样此类物品，会使得模型误判用户的兴趣边界。因此，负采样的另外一个重要目标就是防止采样到伪负例。最普遍方法是使用附加信息来避免采样伪负例。这些附加信息对于提取负信号非常直观，例如社交网络中用户的连接关系^[68,69]、用户的地理位置^[70,71]，以

及额外的交互数据，如已查看但未点击的数据^[4,72]，但是这些额外信息通常不好获得，只有特定的数据集才有。

为了避免采样伪负例，还有一类新的方法是使用多个负例中生成虚拟的负样本用于模型训练。例如，Huang 等人^[60]提出通过混合候选负样本的嵌入来合成虚拟的困难负例。Jun 等人^[73]和 Park 等人^[59]设计了生成对抗神经网络来生成虚拟的困难负例。合成虚拟负样本的方法也广泛使用在计算机视觉中，例如 Zhu 等人^[74]提出了在特征空间上将负样本对应的嵌入插值，类似于构造了虚拟的负样本，而对正样本的嵌入外推，最终提高了对比学习的效果。Zhong 等^[75]提出了一种结合图像混合(Mixup)技术的困难样本构造方法，首先使用 Mixup 合成虚拟的负样本，然后依据合成负样本的相似度分数选择困难的合成负样本进行对比学习训练。

总体来说，负采样方法取得较大进展，特别是困难负采样方法对于下游任务性能提升效果非常显著。采样困难样本是容易实现的，只需要采样相似度分数高或者距离近的样本。但是如何从困难样本中采样困难负样本，还是严重依赖额外信息作为监督信号。尽管合成虚拟负样本的方法看似避免了采样伪负例，但是它的实质是将对比损失函数进行修改：虚拟样本嵌入是若干个负例的混合，那么虚拟样本的相似度分数是这几个负例相似度分数的函数，更多的样本用于合成虚拟样本，导致更多的样本包含在对比损失的分母中，并没有解决伪负样本问题。

2.2.3 正样本数据增强相关工作

数据增强的目标是获取语义不变的可靠正样本用于对比。数据增强是获取语义不变的可靠正样本的主要途径，也是提升自监督对比学习效果最重要的手段，是对比学习的重要组成部分。在传统的图像中，通常的做法是通过在锚点 x 上进行某些语义不变的数据增强来获得正样本 x^+ ，如随机裁剪和翻转^[7]，图像旋转^[76]，和颜色失真^[77]等。然而，仅采用单一的图像增强方法并不利于性能提升，SimCLR^[8]方法综合比较了一系列图像变换方法的效果，并发现将随机裁剪和颜色失真组合起来的变换方式能够获得更好的效果。

受到图像领域的启发，在推荐领域的正样本数据增强工作也得到了广泛的研究。在推荐中数据增强的核心思想是在原有图结构的基础上略作扰动，并以此增强后的新图产生一组新的表征向量，然后将这组新的表征向量与原图产生的表征向量进行对齐，并将不属于同一结点的表征向量互相推远。然而，推荐系统数据集通常以用

户物品二分图的形式存在，节点只包含 ID，往往使用基于图的策略进行增强。图数据的增强策略可以分为以下几种（1）基于特征的增强^[78–82]：仅对节点或者边的特征矩阵进行变换。通常随机屏蔽节点或边属性的一小部分，并用常数或随机值替换。（2）基于结构的增强^[83–86]：仅对邻接矩阵进行变换。通常随机（或手动）在图中添加或删除一小部分边，包括边扰动^[83,84]、节点插入^[85]和边扩散^[86]、图采样^[83]等方法。（3）基于标签的增强^[45,87,88]：由于图上人工标注标签的不足，标签增强是图数据增强研究中的另一个重要方向，用于增加有限的标记训练数据。它可以分为两类：一类是伪标签的增强，例如 BPR^[45] 将未标记样本标记为负例，即是对潜在的用户物品二分图进行了标签增强。第二类是数据混合^[87,88]，直接插值构建虚拟的训练样本。VGCL^[44] 综合考虑了特征增强和结构增强的缺陷，利用变分图重构来估计每个节点的高斯分布，然后从估计分布中进行多个采样来，生成多个对比视图。LightGCL^[28] 则将归一化的邻接矩阵进行奇异值分解，通过抛弃被认为是噪声的较小奇异值，获得了增强视角的图用于对比。

由于图数据具有固有的非欧几里得特性，很难将图像中的随机剪裁、反转、颜色失真等保持语义不变的数据增强方法应用于图数据。基于随机过程的图数据增强可能会损失图中重要的结构信息，因为随机丢弃的点或者边可能会体现图的重要结构信息。此外，数据噪声和数据稀缺问题仍然是获取图数据增强面临的问题。

2.2.4 损失函数设计的相关工作

损失函数是实现对比学习“拉近同类正样本、推远不同类负样本”的关键，是对比学习的核心组件。对比损失的核心思想是通过优化变量互信息，激励编码器编码不同类别样本之间的差异信息，而非样本的像素级信息。鉴于直接最大化互信息很困难，InfoNCE^[7] 率先提出将最大化互信息转化为预测真实分布与合成分布概率密度比的问题，从而实现间接优化互信息。尽管 BPR^[1]、NCE^[24] 是 InfoNCE 损失负例个数为 1 的特例，且工作远早于 InfoNCE，但是优化互信息的概念由 InfoNCE 率先提出。InfoNCE 也广泛应用于推荐领域，文献^[27] 证明了优化 InfoNCE 损失等价于优化 NCDG 指标。本节以更具一般性的 InfoNCE 损失为主线，梳理损失函数设计的相关工作。这些针对 InfoNCE 的改进可以很容易类比到负例个数为 1 的 BPR 损失。

2.2.4.1 面向正负样本概念拓展的损失函数改进

为了获取更密集的数据增强，SimCLR^[8]将同一个样本增强两次，那么同一个批次内的 N 个样本经过两次数据增强就得到了 2N 个样本。同一个样本的两次增强 z_i, z_j 互为正例对，构造了如下对比损失

$$\mathcal{L} = -\log \frac{\exp(f(z_i)^T f(z_j))}{\sum_{k=1}^{2N} \mathbb{I}(k \neq i) \exp(f(z_i)^T f(z_k))}$$

其中分子是同一个样本的两个增强视图的相似度，分母是同一个样本的两个增强视图的相似度加上与其它图像两个增强视图的相似度，合计 2N-1 项。可以看到与 InfoNCE 的数学形式是一致的，只是拓宽了正负样本对的概念。MOCO^[9]变化在于负样本是一个提取的特征组成的队列，损失的计算方式与 InfoNCE 相同。类似地，在推荐中通过一些假设将正负样本对的概念进行拓展，构造了更具细粒度的正负样本对，优化 BPR 损失。例如被购买的物品比只浏览的物品更受用户喜欢^[17]，多次交互的项目比只有一次交互的项目更受喜欢^[89]；用户未看到的潜在项目应该比曝光给用户但没有交互的物品更受喜欢^[14,90,91]。此外，LightGCL^[28]将不同视图下的相同用户视为正样本对，不同用户视为负样本对，优化 InfoNCE 损失。HPM^[92]进一步地通过物品属性信息(如项目类别)构造正样本对。在损失函数分析中，通常不区分由于正负样本对的概念拓展造成的损失函数与初始对比损失函数的差异。

2.2.4.2 面向模型坍塌问题的损失函数改进

由于对比损失优化正例对齐，神经网络的参数空间存在一种可能的解，它为所任意样本映射为同一个相同的特征表示，这样的解也称坍缩解或者模型坍塌。尽管负例参与学习训练有助于分散特征(scatter feature)，避免模型坍塌，但也不能完全避免这种情况。基于这种情况 DirectCLR^[93]取特征向量的前 d 维的子向量，计算对比损失

$$\mathcal{L} = -\log \frac{\exp(f(z'_i)^T f(z'_j))}{\sum_{k=1}^{2N} \mathbb{I}(k \neq i) \exp(f(z'_i)^T f(z'_k))}$$

其中， $z' = z[0 : d]$ 是原始特征向量的前 d 个维度构成子向量，这个对比损失与 SimCLR 损失的唯一区别就是只取每个特征向量的前 d 维。

BYOL^[10]进一步改进了对比损失。对同一个样本的两次增强，两个网络分支学

到的归一化特征表示分别记为 $\bar{q}(z)$ 和 \bar{z}^I , BYOL 优化归一化特征表示的均方误差:

$$\|\bar{q}(z) - \bar{z}^I\|^2 = 2 - 2 \cdot \frac{\langle q(z), z^I \rangle}{\|q(z)\|_2 \cdot \|z^I\|_2} = 2 - 2 \cos(q(z), z^I)$$

其中为 $q(z), z^I$ 分别为两个网络分支学到的特征表示向量 (未归一化)。可以看到, BYOL 实际上只约束两个网络分支学到的特征表示的方向一致, 并不规定正例对表示向量的具体模值。除了损失函数外, BYOL 还通过精致的网络架构设计和正则化技术分散特征, 避免神经网络学到坍缩解。

2.2.4.3 面向梯度消失问题的损失函数改进

InfoNCE 损失函数的形式决定了其正例梯度等于所有负例梯度的和, 符号相反。对于一个分数较小的负样本(容易负样本), 或者分数较大的正样本(容易正样本), 梯度容易消失, 使得模型从样本中学到较少的信息。具体到负例个数为 1 的 BPR 损失中, 就有正例的梯度等于负例的梯度, 同样符号相反, 使得梯度消失的问题更加明显, 因此往往需要通过保持较大的负例个数 N, 以避免梯度消失的问题。于是解耦对比损失^[94] (Decoupled Contrastive Learning) 去掉了 InfoNCE 失函数分母中两个正样本对之间的相似度, 损失函数的形式计算如下:

$$\mathcal{L} = -\log \frac{\exp(f(x)^T f(x^+))}{\sum_{i=1}^N \exp(f(x)^T f(x^-))}$$

解耦对比损失在一定程度上解决了容易正样本和容易负样本所导致的梯度消失问题, 也避免了负样本数量对梯度的放缩。

2.2.4.4 面向自监督场景的损失函数改进

在自监督场景下伪负例的存在, 使得 InfoNCE 损失实际上优化实例之间的对齐和均匀, 而非不同类别样本的均匀与对齐。更具体地, 由于伪负例包含其中, 在自监督场景下, InfoNCE 实际上是拉近同一个样本的增强之间的距离, 推远不同的样本而非不同类别的样本之间的距离, 因为任意锚点以外的样本都被视作负样本。负例的无监督, 使得自监督场景下 InfoNCE 偏离了原始的优化目标, 对下游泛化性能产生了不利的影响。为了鼓励编码器去编码数据的语义结构, 拉近同类样本, 推远不同类样本, ProtoNCE^[95] 通过改进正负样本对的语义, 以改进对比损失。具体地, 当前样本的特征与其对应的聚类中心构成正样本对, 而与其他聚类中心形成负样本

对，从而得到如下优化目标：

$$\mathcal{L} = -\log \frac{\exp(f(x)^T f(c_i))}{\sum_{i=1}^C \exp(f(x)^T f(c_i))}$$

其中 c_i 为聚类中心，也被称为原型 (prototypes)，通过在 EM 算法的期望步骤中执行 k-means 聚类得到。通过将分子修改为样本与所属类别中心的相似度，分母部分更改为样本与其他聚类中心之间的相似度，实现将优化目标改进为拉近样本与所属类别中心的距离，推远样本与其它类别中心的距离。这一思想在个性化推荐中也有应用^[96]。

上述方案需要迭代计算，开销较大。另外一个解决负例无监督的途径是通过校正负样本相似度分数，以获取一个与有监督对比损失一致的改进损失函数，从而取得类似于监督学习的泛化效果，代表工作是去偏对比学习^[97] (Debiased Contrastive Learning, DCL)。其核心思想是，由于伪负例包含在 InfoNCE 的分母中，导致的结果是分母中的负例相似度分数计算不准确。可以通过修正负例的相似度分数，实现对有监督对比损失的近似。其损失函数计算如下

$$\mathcal{L} = -\log \frac{\exp(f(x)^T f(x^+))}{\exp(f(x)^T f(x^+)) + N \cdot g}$$

其中 g 是对真负样本与锚点相似度分数期望值的估计，其具体形式在后面的章节中会有更详尽的介绍。

在 DCL^[97] 伪负例去偏的基础上，HCL^[98] 进一步地考虑了硬负例挖掘，即将困难负样本推远离锚点。它遵循 DCL 的框架，并通过如下方式进一步地加权未标注样本的相似度分数：

$$\omega_i^{\text{HCL}} = \frac{\hat{x}_j^\beta}{\frac{1}{N} \sum_{j=1}^N \hat{x}_j^\beta}$$

其中 \hat{x} 是未标注样本与锚点的相似度分数， β 是超参数。因此，未标注样本的相似度分数越高，其分配到的权重就越大，导致其被推的越远。基于估计校正的损失函数改进，避免了额外的计算开销，取得了很好的效果。

2.3 预备知识

对于深度神经网络而言，准确推导其输出的相似度分数的分布是很困难的，本文会频繁使用来自样本计算得到的经验分布函数来近似。此外，个性化推荐是一个

排序任务，次序统计量作为排序分析的一个有力工具，本文也会频繁使用到。最后，成对学习是极大似然估计量，也会频繁使用到。因此本节对这三个会频繁用到的预备知识进行介绍。

2.3.1 经验分布函数

在介绍经验分布函数之前，先回顾累积分布函数 (cumulative distribution function, CDF) 的定义。对于一个实值随机变量 X ，累积分布函数 (CDF) 在 x 处的值表示随机变量 X 取值小于或等于 x 的概率

$$F(x) = \mathbb{P}(X \leq x)$$

$F(x)$ 满足 $\lim_{x \rightarrow -\infty} F(x) = 0$ 且 $\lim_{x \rightarrow \infty} F(x) = 1$ 。

对于连续随机变量 X ，其累积分布函数 CDF 可以表示为其概率密度函数 f_X 的积分，如下所示：

$$F(x) = \int_{-\infty}^x f_X(t) dt$$

因此，只要给出概率密度函数 $f_X(x)$ 的表达式，就可以准确计算累积分布函数 $F(x)$ 。然而在绝大部分场景下，获取概率密度函数 $f_X(x)$ 的表达式是困难的，那么累积分布函数 $F(x)$ 的计算通常依赖于经验分布函数。

经验分布函数，通常也称为经验累积分布函数 (empirical cumulative distribution function, eCDF) 是与样本的经验观测相关联的分布函数。在观测变量的任何指定值处，经验分布函数定义为小于或等于指定值的观测变量的比例：

$$\hat{F}(x) = \frac{1}{N} \sum_{i=1}^N \mathbb{I}(X_i \leq x) \quad (2.4)$$

其中， $\mathbb{I}(\cdot)$ 是指示函数。以一个例子说明经验分布函数的计算。假设有五个独立同分布的观测值 $\{3, 7, 2, 6, 4\}$ ，那么可以计算得到 $\hat{F}(1) = 0$ ，表明五个观测值小于等于 1 的占比为 0； $\hat{F}(2) = 1/5$ ，表明五个观测值小于等于 2 的占比为 $1/5$ ； $\hat{F}(3) = 2/5$ ，表明五个观测值小于等于 3 的占比为 $2/5$ 。以此类推， $\hat{F}(100) = 1$ ，表明五个观测值小于等于 100 的占比为 1。

经验分布函数 $\hat{F}(x)$ 是标准的累积分布函数 $F_X(x)$ 的估计^[99]。对于固定的 x ，指示函数 $\mathbb{I}(X_i \leq x)$ 是一个参数为 $p = F(x)$ 的伯努利随机变量。因此， $n\hat{F}(t)$ 是一个具

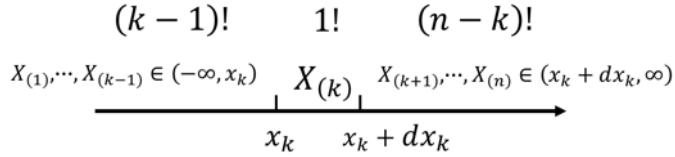


图 2.2 次序统计量的概率微元

有均值 $nF(t)$ 和方差 $nF(t)(1 - F(t))$ 的随机变量。这意味着 $\hat{F}(t)$ 是 $F(t)$ 的无偏估计量。此外，根据 Glivenko 定理^[100] 证明了，经验分布函数一致收敛于分布函数。

在很多场景下，想获取累计分布函数 (CDF) $F(x)$ 的解析表达式是非常困难的，但是计算经验分布函数 (eCDF) 却非常容易实现。于是可以通过计算经验分布函数 (eCDF)，以实现对抽象的分布函数的近似。经验分布函数是一个强有力的工具，通过少数观测样本对难以获取的分布函数进行近似，实现对总体分布特征的刻画，本文会多次使用这个工具。

2.3.2 次序统计量

对于 n 个独立同分布随机变量 X_1, X_2, \dots, X_n ，根据其取值升序排列，有

$$X_{(1)} \leq X_{(2)} \leq \dots \leq X_{(n)}$$

那么称 $X_{(k)}$ 为排序第 k 的次序统计量^[101]。下面简单分析如何求解次序统计量 $X_{(k)}$ 的概率密度表达式。

记随机变量 X 的概率密度为 $f(x)$ ，相应的累积分布函数记为 $F(x) = \int_{-\infty}^x f(t)dt$ ，可以使用概率微元法^[99] 求得次序统计量 $X_{(k)}$ 的概率密度表达式。

对于随机变量的 n 个观测值 $\{x_i\}_{i=1}^n$ ，一定存在一个足够小的区间 dx_k ，使得有且仅有一个随机变量 $X_{(k)}$ 的取值 $x_k \in [x_k, x_k + dx_k]$ ，如图2.2所示。由于次序统计量 $\{X_{(i)}\}_{i=1}^n$ 是根据它们的取值升序排列，那么前 $k-1$ 个随机变量 $\{X_{(i)}\}_{i=1}^{k-1}$ 的取值 $\{x_i\}_{i=1}^{k-1} \in (-\infty, x_k)$ ，剩余的 $n-k$ 个随机变量 $\{X_{(i)}\}_{i=k+1}^n$ 的取值 $\{x_i\}_{i=k+1}^n \in (x_k + dx_k, \infty)$ 。 $X_{(k)}$ 的概率微元可以计算为

$$g(x_k; k, n)dx_k = \frac{n!}{(k-1)!(n-k)!}[F(x_k)]^{k-1}f(x_k)dx_k[1 - F(z_k)]^{n-k} + o(dx_k)$$

$g(x_k; k, n)$ 即为次序统计量 $X_{(k)}$ 的概率密度。其中， $[F(z_k)]^{k-1}$ 是前 $k-1$ 个随机变

量落入区间 $(-\infty, x_k)$ 的概率; $[1 - F(x_k)]^{n-k}$ 是后面 $n - k$ 个变量落入区间 (x_k, ∞) 的概率, 因为 $1 - F(x_k) = P(X > x_k)$ 。 $o(dx_k)$ 是 dz_k 的高阶无穷小量。将上式两边同时除以 dz_k , 则可以获得次序统计量的密度表达式

$$g(x_k; k, n) = \frac{n!}{(k-1)!(n-k)!} [F(z_k)]^{k-1} f(x_k) [1 - F(x_k)]^{n-k}$$

上式意味着, 只要给出随机变量的分布 $f(x)$, 就可以写出对应的次序统计量分布 $g(x)$ 。

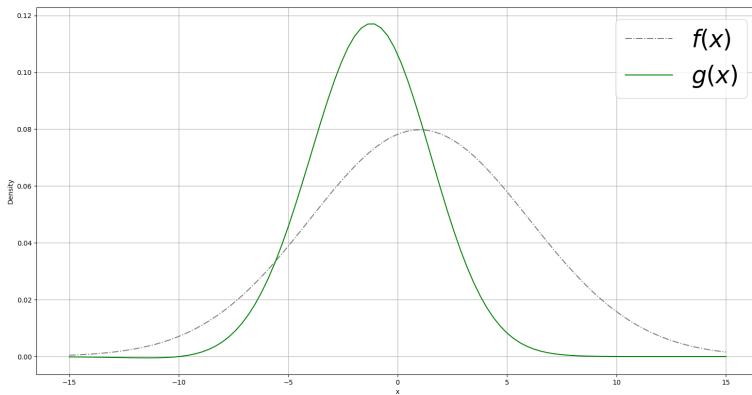


图 2.3 随机变量及其次序统计量分布示意图

图2.3给出了正态分布 $N(1, 5)$ 的随机变量对应的次序统计量分布 $g(x; k = 2, n = 5)$ 的示意图。可以看到, 次序统计量的分布 $g(x)$ 比 $f(x)$ 更窄。可以使用一个例子理解这一现象: 假设学生考试成绩 X 独立同分布于 $f(x)$ 。那么在一个班级中排名最后的学生的成绩的分布可以推导为 $g(x)$, 其可能的成绩分数比分布 $f(x)$ 更窄。这是因为“排名最后的学生”这个信息减少了随机变量 X 的不确定性。

如果随机变量 X 中, 只包含两个总体, 例如正例总体和负例总体, 那么可以取 $n = 2$, 得到次序统计量 $X_{(1)}$ 和次序统计量 $X_{(2)}$ 的密度为:

$$g(x_1) = 2f(x)[1 - F(x)] \quad (2.5)$$

$$g(x_2) = 2f(x)F(x) \quad (2.6)$$

次序统计量是非参数统计推断的最基本工具, 广泛应用于诸多统计理论与实践, 如分位数估计、极值分析等、排序分析等。由于个性化推荐是一个排序任务, 因此次序统计量也为推荐算法提供了重要的工具。上式的两个结论, 会在后面的章节多

次应用。

2.3.3 极大似然估计

极大似然估计（MLE）是一种估计参数化的概率分布的方法。其基本思想是，在参数空间内，求解一个模型参数 Θ ，使得在一次实验中观测到观测值（也称为证据） X 的概率最大。这通过最大化似然函数来实现：

$$\arg \max p(X|\Theta)$$

其中， $p(X|\Theta)$ 称为似然，反映了模型参数 Θ 对观测证据 X 的解释程度^[99]。以 BPR 为例，简单介绍极大似然估计的运作原理。给定观测到的正负样本对 X ，用户“偏好正例 i 强于负例 j ”的似然被建模为

$$p(X|\Theta) = -\log \sigma(\hat{x}_{ui} - \hat{x}_{uj})$$

其中 $\hat{x}_{ui}, \hat{x}_{uj}$ 是由用户物品特征表示 Θ 参数化的偏好预测值。那么 $p(X|\Theta)$ 描述了当前模型参数下，观测到用户偏好物品 i 强于 j 的概率。引入样本的独立性假设后，多个样本的似然函数是多个单个样本似然的乘积。通过对数运算将乘积运算转化为求和运算，原始的极大似然估计问题往往转化为最大化对数似然问题。

在实践中，由于要引入正则化项避免过拟合，而正则化项又等价于高斯分布先验密度的对数，因此正则化项为 BPR 的估计量提供了后验概率解释。将 BPR 解释为极大似然估计，还是最大后验估计，主要是对正则化项的语义理解不同：若狭义地理解正则化项，那么 BPR 的估计量即为极大似然估计量；若把正则化项理解为高斯先验密度的对数，那么 BPR 为最大后验估计量。在本文成对学习的具体环境中，不区分由正则化项的语义造成的极大似然或者最大后验估计的概念差异。从贝叶斯推断的角度来看，极大似然估计通常等同于具有均匀先验分布（或标准差为无穷大的正态先验分布）的最大后验估计（MAP）估计^[102]。

3 正例去噪算法研究

对于个性化推荐应用而言,最广泛使用的隐式反馈是一种不完全数据(incomplete data),只能观察到交互的有无,而无法观察到该交互所对应偏好的大小。隐式反馈数据中通常包含一些不可信交互(误报),即伪正例,例如代购、误触等交互数据。由这些伪正例构建的成对比较,是不可信的成对比较,也称为含噪声成对比较,导致成对学习模型的不准确优化。本章提出一种自适应去噪的成对排序算法(Bayesian Personalized Ranking with Autonomous Credence, BPRAC),用于从含噪成对比较的数据中学习个性化排序。由于交互对应的偏好值不可观测,本章引入一个隐变量作为衡量交互置信度的新指标,这个隐变量与用户物品表示一起作为模型参数进行端到端地学习。所提出的BPRAC算法采用期望最大化框架:在期望步骤中使用贝叶斯推断来估计交互的置信度指标;在最大化步骤中固定置信度指标,更新参数学习用户和物品的表示。在真实世界的数据集上进行的实验证实自适应去噪的成对排序算法的优越性。

3.1 引言

噪声的存在会对机器学习模型的性能和泛化能力产生负面影响。首先,噪声可能导致模型对数据的过拟合。模型可能会将噪声视为真实模式,从而导致错误的学习和预测。其次,噪声可能降低模型的准确性和稳定性。如果训练数据包含大量噪声,模型可能无法准确地捕捉数据的真实分布和模式,从而导致预测结果的不可靠性。噪声在机器学习中广泛存在,可以来自多个方面,包括数据收集过程中的错误、不准确的标注、不完整的数据、异常值等。隐式反馈数据是典型的不完全数据,只能观测到用户的交互记录,但是无法观测到交互所对应偏好的高低。隐式反馈数据中存在一些用户的手滑、误触、代购等无偏好的交互行为,会被记录并当作正例,构成了伪正例问题。

如图3.1所示,一个运动爱好者在节日期间给女友购买了一个口红 i_3 ,这是一个不体现该用户偏好的偶然交互行为。成对比较是由已经交互的正例和未交互的负例构成,由于 i_3 是伪正例,任何包含 i_3 的成对比较,都是噪声成对比较。如果不能消

除这些噪声成对比较的影响，推荐模型会误认为该运动爱好者更偏好口红而非其他运动物品，从而产生一系列例如女性用品的错误推荐结果。

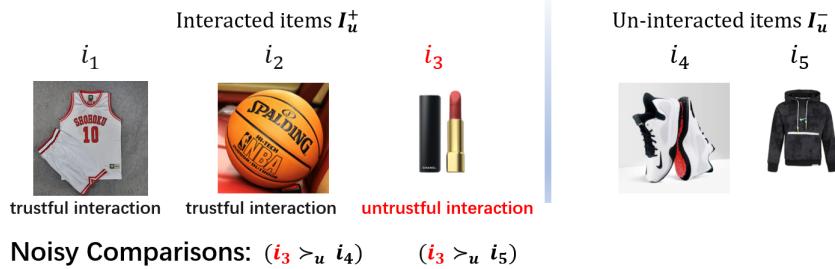


图 3.1 含噪成对比较示意图

一般而言，存在两种类型的噪声：一种是特征依赖的噪声^[103]，例如标注者可能倾向于混淆狼和狗，但不会混淆老虎和狗，因此狼容易被错误标注而老虎不会；另外一种是独立于特征的噪声^[104]，如高斯白噪声。推荐系统的误报兼具二者特征。例如误触，与物品的特征无关，每个交互的噪声率是随机均匀分布的；而代购，则与物品的特征有关，具有礼物性质的物品更容易产生较高的噪声率。

在机器学习的基础理论方面，先前的研究解决标签噪声学习主要是从数据、目标函数和优化策略的角度出发，并提出了相应的解决方法和技术。具体而言，从数据的角度，重点是建立并估计噪声转移矩阵，该噪声转移矩阵模拟了标签错误的过程，可以用于还原数据的原始的所属类别^[105]。这个噪声转移矩阵还可以用来校正损失，构建与干净数据风险一致的估计^[106]。从目标函数的角度出发，主要思路是经验风险重写（Empirical Risk Rewriting）。例如文献^[107]通过重加权样本设计能够容忍噪声的损失函数，它可以比标准损失函数更具鲁棒性。最后，从优化策略的角度出发，重点是探索优化的动态过程，以解决标签噪声学习问题。核心在于，神经网络倾向于先拟合干净数据，然后拟合噪声数据^[108]。基于这一发现，例如早停法^[109]作为一种简单而有效的方法，用于避免在噪声数据上过拟合；而 Co-teaching 方法^[110]通过小损失技巧相互筛选干净样本，鲁棒地同时训练两个网络。

在成对学习的问题设置下，噪声问题更具挑战性。例如从优化策略出发的“神经网络倾向于先拟合干净数据，然后拟合噪声数据”的经验性观察主要是基于单个样本构建的损失，是否在成对样本构建的损失上依旧起作用是存疑的，因为在推荐中往往是难以拟合的困难样本对性能的提升贡献越大^[45]。此外，由于推荐系

统中的噪声既具有特征依赖性又具有特征无关性，使用噪声转移矩阵模拟噪声标签的生成过程或估计权重以对样本进行重加权都变得非常具有挑战性。回溯伪正例的本质，是隐式反馈数据的特性造成的：它是一种不完全数据，无法观测到交互所对应的偏好值，才产生了噪声成对比较。如果能观测到每个交互偏好强度的高低，那么根据偏好值大小构造的成对比较就是可靠的。从统计视角，期望-最大化(Expectation-Maximization)最大化框架是一个从不完全数据中进行极大似然或者最大后验估计非常有效的方法，可以估计无法观测到的隐变量和模型参数。因此，本章引入一个隐变量，含义为每个交互的偏好值，同时也对应了每个成对比较置信度的高低，交互的偏好值越大则成对比较的置信度越高。从噪声成对比较中学习排序则转化为一个含有隐变量的最大后验估计问题：在期望步骤中使用贝叶斯推断来估计置信度指标；在最大化步骤中固定置信度指标，更新模型参数学习用户和物品的表示，以减少噪声成对比较对学习算法的影响，学习更加准确的用户和物品表示。

3.2 自适应去噪的成对排序算法

3.2.1 问题设置

设 $\mathbf{X} = [x_{ui}] \in \mathbb{R}^{M \times N}$ 表示包含 M 个用户和 N 个物品的交互矩阵，其中 $x_{ui} \in \{1, 0\}$ 。令 \mathcal{U} 和 \mathcal{I} 分别表示用户集合和物品集合。对于用户 $u \in \mathcal{U}$ ，令 $\mathcal{I}_u^+ \subseteq \mathcal{I}$ 表示他交互过的物品集合， $\mathcal{I}_u^- = \mathcal{I} \setminus \mathcal{I}_u^+$ 表示他未交互过的物品集合。通过选择一个属于 \mathcal{I}_u^+ 的物品 i 和一个属于 \mathcal{I}_u^- 的物品 j ，可以构建成对比较的实例。训练数据集可以通过以下方式构建：

$$\mathcal{D} = \{(u, i, j) | i \in \mathcal{I}_u^+ \wedge j \in \mathcal{I}_u^-\}. \quad (3.1)$$

BPR 的目标是从训练数据集中学习用户和物品表示 Θ ，使得观测到 \mathcal{D} 的后验概率最大化，即：

$$\mathcal{L}_{\text{BPR}} = \prod_{(u, i, j) \in \mathcal{D}} P(i \succ_u j | \Theta) P(\Theta). \quad (3.2)$$

如果一个交互 $x_{ui} = 1$ 是噪声（伪正例），则所有包含 i 的成对比较 $i \succ_u j$ 即为噪声比较，它们描述了不准确的用户偏好结构。这些噪声成对比较不应该包含在最大后验目标之内，否则会学到不准确的用户物品表示。然而，由于无法观测哪些交互是

可信的，因此引入新的参数 c_{ui} 作为隐变量来衡量成对比较 $i \succ_u j$ 的置信度。具体而言，对于一个可信的交互， $c_{ui} = 1$ ；否则， $c_{ui} = 0$ 。那么可以修正 BPR 的优化目标为：

$$\mathcal{L}' = \prod_{(u,i,j) \in \mathcal{D}} [P(i \succ_u j | \Theta) P(\Theta)]^{c_{ui}}. \quad (3.3)$$

其中， c_{ui} 是隐藏变量，对交互项 (u, i) 进行加权。当 $c_{ui} \rightarrow 0$ （即不可信的交互）时，噪声比较 $i \succ_u j$ 的后验概率被固定为 1，用户和物品表示 Θ 的更新不受该噪声成对比较的影响。作为一个可学习的变量，隐变量 c_{ui} 提供了从噪声数据中进行表示学习的解决方案。从含噪成对比较中学习排序的问题，可以形式化为一个含有隐变量的最大后验估计问题。

跟随 BPR 论文的符号规则，使用 \hat{x}_{ui} 来表示用户 u 对物品 i 的偏好预测值。用户偏好物品 i 胜过物品 j 的似然为：

$$P(i \succ_u j | \Theta) = \sigma(\hat{x}_{ui} - \hat{x}_{uj}), \quad (3.4)$$

其中， $\sigma(\cdot)$ 是 Sigmoid 函数： $\sigma(z) = \frac{1}{1+e^{-z}}$ 。将公式(3.4)带入公式(3.3)中，并两边取对数，得到了如下最终的优化目标：

$$\ln \mathcal{L}' = \sum_{(u,i,j) \in \mathcal{D}} c_{ui} [\ln \sigma(\hat{x}_{ui} - \hat{x}_{uj}) - \lambda \|\Theta\|^2], \quad (3.5)$$

其中， $\lambda \|\Theta\|^2$ 是模型训练中的正则化项。下面介绍在矩阵分解模型下，如何求解这个含有隐变量的最大后验估计问题。

3.2.2 先验分布

BPR 论文设定了用户物品表示为两个 d 维高斯随机向量的内积，基于矩阵分解得到的相似度分数先验分布由如下引理给出。

引理 3.1： 设 $x = \langle \mathbf{w}, \mathbf{h} \rangle$ 表示两个 d 维随机向量的内积，其中 \mathbf{w} 和 \mathbf{h} 的元素是独立同分布的随机变量，每个随机变量都服从 $\mathcal{N}(0, \lambda)$ 的高斯分布。那么 x 的概率密度函数 (PDF) 为：

$$f_X(x; d, \lambda) = \frac{1}{\lambda \sqrt{\pi} \Gamma(\frac{d}{2})} \left(\frac{|x|}{2\lambda} \right)^{\frac{d-1}{2}} K_{\frac{d-1}{2}} \left(\frac{|x|}{\lambda} \right), \quad (3.6)$$

其中， $K_\eta(\cdot)$ 是第三类修正的贝塞尔函数， $\Gamma(\cdot)$ 是伽玛函数。

华 中 科 技 大 学 博 士 学 位 论 文

证明： 随机变量 $x = \langle \mathbf{w}, \mathbf{h} \rangle$ 可以通过以下代数运算转化为两个伽玛分布的随机变量的差：

$$\begin{aligned} x &= \langle \mathbf{w}, \mathbf{h} \rangle = \sum_{k=1}^d w_k \cdot h_k \\ &= \frac{\lambda}{2} \sum_{k=1}^d \left[\frac{(w_k + h_k)^2}{2\lambda} - \frac{(w_k - h_k)^2}{2\lambda} \right] \end{aligned} \quad (3.7)$$

那么

$$\frac{2}{\lambda} x = \sum_{k=1}^d \frac{(w_k + h_k)^2}{2\lambda} - \sum_{k=1}^d \frac{(w_k - h_k)^2}{2\lambda}, \quad (3.8)$$

其中 $\frac{(w_k + h_k)}{\sqrt{2\lambda}} \sim \mathcal{N}(0, 1)$, $\frac{(w_k - h_k)}{\sqrt{2\lambda}} \sim \mathcal{N}(0, 1)$. 于是第一项 $z_1 = \sum_{k=1}^d \frac{(w_k + h_k)^2}{2\lambda}$ 和第二项 $z_2 = \sum_{k=1}^d \frac{(w_k - h_k)^2}{2\lambda}$ 服从自由度为 d 的卡方分布。注意到卡方分布 $\chi^2(d)$ 是参数为 $Ga(z; \frac{d}{2}, \frac{1}{2})$ 的伽玛分布族的特例, 因此变量 Z_1 and $-Z_2$ 的特征函数为:

$$\phi_{Z_1}(t) = (1 - 2it)^{-\frac{d}{2}}, \quad (3.9)$$

$$\phi_{-Z_2}(t) = (1 + 2it)^{-\frac{d}{2}}, \quad (3.10)$$

其中 i 为虚数单位。进一步地, 记随机变量 $Y = \frac{2}{\lambda} X = Z_1 + (-Z_2)$, 则随机变量 Y 的特征函数 $\phi_Y(t)$ 是 $\phi_{Z_1}(t)$ 和 $\phi_{-Z_2}(t)$ 的乘积:

$$\phi_Y(t) = \phi_{Z_1}(t) \cdot \phi_{-Z_2}(t) = (1 + 4t^2)^{-\frac{d}{2}}. \quad (3.11)$$

那么随机变量 Y 的概率密度函数 $f_Y(y)$ 可以通过如下傅里叶逆变换计算得到:

$$\begin{aligned} f_Y(y) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \phi_Y(t) e^{ity} dt \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} (1 + 4t^2)^{-\frac{d}{2}} e^{ity} dt. \end{aligned} \quad (3.12)$$

直接计算上式 $f_Y(y)$ 是复杂的。观察到特征函数 $\phi_Y(t)$ 与方差伽玛 (VG) 分布^[111,112] 的特征函数具有相同的函数形式。VG 分布的特征函数为

$$\phi_{VG}(t) = E(e^{itX}) = e^{i\delta t} (1 - i\theta vt + \frac{\sigma^2 v}{2} t^2)^{-1/v}. \quad (3.13)$$

VG 分布的概率密度函数 $f_{VG}(y)$ 为:

$$f_{VG}(y; \theta, \delta, \sigma, v) = \frac{2e^{\frac{\theta(y-\delta)}{\sigma^2}}}{\sigma\sqrt{2\pi}v^{\frac{1}{v}}\Gamma(\frac{1}{v})} \left(\frac{|y-\delta|}{\sqrt{\frac{2\sigma^2}{v} + \theta^2}} \right)^{\frac{1}{v}-\frac{1}{2}} \cdot K_{\frac{1}{v}-\frac{1}{2}}\left(\frac{|y-\delta|\sqrt{\frac{2\sigma^2}{v} + \theta}}{\sigma^2} \right). \quad (3.14)$$

取 $\delta=0, \theta=0, v=2/d, \sigma=2\sqrt{d}$, 那么

$$\phi_{VG}(t) = (1 + \frac{1}{4}t^2)^{-\frac{d}{2}} = \phi_Y(t), \quad (3.15)$$

由于特征函数与概率密度函数是一一对应的关系, 因此 Y 的概率密度函数为

$$\begin{aligned} f_Y(y) &= f_{VG}(y; \theta = 0, \delta = 0, \sigma = 2\sqrt{d}, v = \frac{2}{d}) \\ &= \frac{1}{\sqrt{2\pi d}(\frac{2}{d})^{\frac{d}{2}}\Gamma(\frac{d}{2})} (\frac{|y|}{2d})^{\frac{d-1}{2}} K_{\frac{d-1}{2}}(\frac{|y|}{2}) \\ &= \frac{1}{\sqrt{2\pi} 2^{\frac{d}{2}} \Gamma(\frac{d}{2})} (\frac{|y|}{2})^{\frac{d-1}{2}} K_{\frac{d-1}{2}}(\frac{|y|}{2}). \end{aligned} \quad (3.16)$$

需要注意的是, $\frac{\lambda}{2}Y = X$ 。通过将 Y 按照常数 ρ 进行缩放, 可以生成一个经过缩放的随机变量分布

$$\rho Y \sim \frac{1}{\rho} f_Y(\frac{y}{\rho}). \quad (3.17)$$

取 $\rho = \frac{\lambda}{2}$, 则 $X = \frac{\lambda}{2}Y \sim \frac{2}{\lambda}f_Y(\frac{2}{\lambda}y)$, 进而可以得到随机变量 X 的概率密度函数为:

$$f_X(x; d, \lambda) = \frac{1}{\lambda\sqrt{\pi}\Gamma(\frac{d}{2})} \left(\frac{|x|}{2\lambda}\right)^{\frac{d-1}{2}} K_{\frac{d-1}{2}}\left(\frac{|x|}{\lambda}\right), \quad (3.18)$$

□

在 BPR 中, 论文假设参数用户和物品的表示 Θ 服从零均值和对角协方差矩阵的多元高斯分布, 那么参数矩阵 Θ 的所有元素都是独立且同分布 (i.i.d) 的随机变量, 每个随机变量都服从 $\mathcal{N}(0, \lambda)$ 的高斯分布。按照如上的多元高斯分布初始化后, 使用引理3.1的结果, 可以得到基于矩阵分解的偏好预测值的分布为:

$$f(\hat{x}_{ui}) = \frac{1}{\lambda\sqrt{\pi}\Gamma(\frac{d}{2})} \left(\frac{|\hat{x}_{ui}|}{2\lambda}\right)^{\frac{d-1}{2}} K_{\frac{d-1}{2}}\left(\frac{|\hat{x}_{ui}|}{\lambda}\right), \quad (3.19)$$

相应的累积分布函数 (CDF) 为:

$$F(\hat{x}_{ui}) = \int_{-\infty}^{\hat{x}_{ui}} f(t)d(t). \quad (3.20)$$

3.2.3 后验分布

考虑 n 个独立同分布的随机变量 $\{Z_i\}_{i=1}^n$, 其对应的取值 $\{z_i\}_{i=1}^n$ 按升序排列: $X_1 \leq X_2 \leq \dots \leq X_n$, 其中 X_k 是取值 x_k 在第 k 个位置的随机变量, 那么 x_k 的概率密

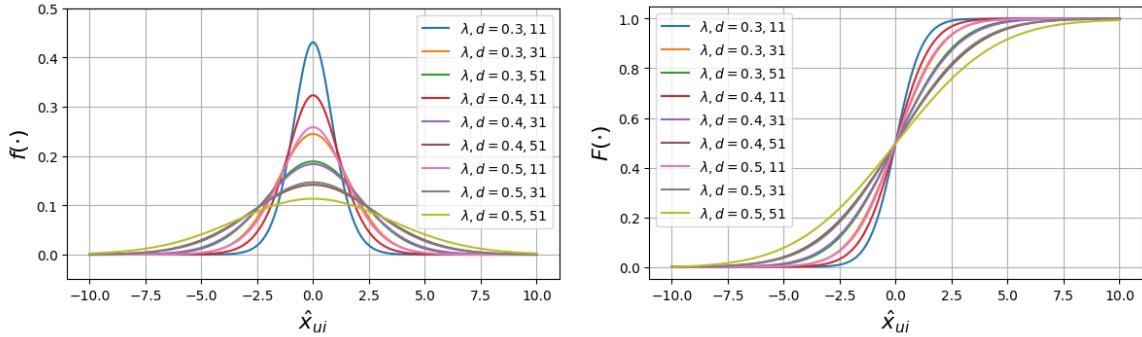


图 3.2 不同参数下先验分布示意图

度为

$$g(x_k; k, n) = \frac{n!}{(k-1)!(n-k)!} [\Phi(x_k)]^{k-1} \phi(x_k) [1 - \Phi(x_k)]^{n-k}. \quad (3.21)$$

对于一个交互，只存在两种可能：(1) 它是真正例，即用户真的喜欢；(2) 它是伪正例，即用户不喜欢。预测评分只存在两个总体，于是取 $n = 2$ 。此外，模型是针对负例评分小于正例评分优化，对于关心的真正例的预测评分 \hat{x}_{ui} ，如果它是用户喜欢的正例，应该满足 $\hat{x}_{uj} \leq \hat{x}_{ui}$ ，这正是次序统计量 $X_{(2)}$ 的取值。于是可以写出真正例的预测评分 \hat{x}_{ui} 的分布

$$\begin{aligned} g(\hat{x}_{ui}; k=2, n=2) &= \frac{2!}{(2-1)!(2-2)!} [F(\hat{x}_{ui})]^{2-1} f(\hat{x}_{ui}) [1 - F(\hat{x}_{ui})]^{2-2} \\ &= 2F(\hat{x}_{ui})f(\hat{x}_{ui}), \end{aligned} \quad (3.22)$$

其中 $f(\cdot)$ 由公式3.19给出， $F(\cdot)$ 由公式3.20给出。图3.2绘制了不同参数下先验分布示意图，图3.3绘制了不同参数下后验分布示意图，可以看到先验分布和后验分布满足非负性和归一性条件。

3.2.4 学习算法

在本节中，介绍了的学习算法并分析了其收敛性。该学习算法在 EM 框架下包括两个步骤：(i) 在期望步骤中估计 c_{ui} ，以及 (ii) 在最大化步骤中学习用户和物品表示 Θ 。

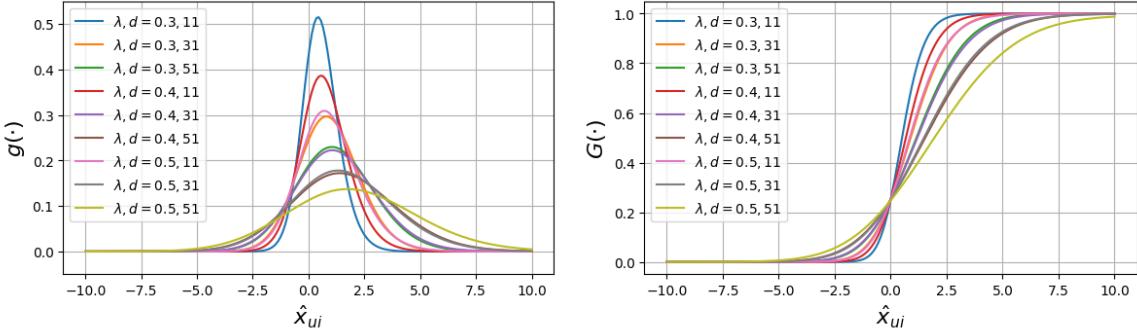


图 3.3 不同参数下后验分布示意图

3.2.4.1 期望步骤

在期望步骤中，固定的模型参数 Θ ，即给定预测评分，目标是估计 c_{ui} 。首先写出 $Q(\cdot)$ 函数为

$$\begin{aligned} Q(\Theta, \Theta^{(t)}) &= \mathbb{E}_{c_{ui}} [\ln P(\Theta | \succ_u, c_{ui}) | \succ_u, \Theta^{(t)}] \\ &= \mathbb{E}_{c_{ui}} [\sum_{(u,i,j) \in \mathcal{D}} c_{ui} (\ln \sigma(\hat{x}_{ui} - \hat{x}_{uj}) - \lambda \|\Theta\|^2)] \\ &= \sum_{(u,i,j) \in \mathcal{D}} \mathbb{E}[c_{ui}] [\ln \sigma(\hat{x}_{ui} - \hat{x}_{uj}) - \lambda \|\Theta\|^2], \end{aligned} \quad (3.23)$$

其中， $\mathbb{E}[c_{ui}]$ 是给定在第 t 次学习迭代中模型参数 $\Theta^{(t)}$ 的条件下， c_{ui} 的期望值。使用 $\mathbb{E}[c_{ui}]$ 作为 c_{ui} 的估计。由于 c_{ui} 是一个二值伯努利变量，有以下关系式：

$$\begin{aligned} \hat{c}_{ui} = \mathbb{E}[c_{ui}] &= P(c_{ui} = 1 | \Theta, \succ_u) \\ &= P(c_{ui} = 1 | \hat{x}_{ui}, \succ_u) \end{aligned} \quad (3.24)$$

$$= \frac{P(\hat{x}_{ui}, \succ_u | c_{ui} = 1)}{P(\hat{x}_{ui}, \succ_u)} P(c_{ui} = 1) \quad (3.25)$$

$$\propto P(\hat{x}_{ui}, \succ_u | c_{ui} = 1) P(c_{ui} = 1) \quad (3.26)$$

$$= g(\hat{x}_{ui}; n = 2, k = 2) P(c_{ui} = 1) \quad (3.27)$$

需要注意的是，当固定模型参数 Θ 时，预测评分 \hat{x}_{ui} 也被固定，于是有公式(3.24)。公式(3.25)是基于贝叶斯公式得到，其中 $P(c_{ui} = 1)$ 是交互为可信的先验概率，分母 $P(\hat{x}_{ui}, \succ_u)$ 在贝叶斯分析中被省略为归一化常数。 $P(\hat{x}_{ui}, \succ_u | c_{ui} = 1)$ 是可信交互的概率密度。给定所有的预测评分，对于一个可信的交互 (u, i) ，其预测评分 \hat{x}_{ui} 应该满足 $\hat{x}_{uj} \leq \hat{x}_{ui}$ ，此时概率密度由公式(3.22)给出。在给定 $P(c_{ui} = 1)$ 的情况下，观测值

\hat{x}_{ui} 在分布 $g(\hat{x}_{ui}; n = 2, k = 2)$ 中对应的概率密度越高，它是可信交互的可能性越大。

设 n_i 表示物品 i 的选择比例，即与该物品进行交互的用户的百分比。令 \bar{n} 和 σ_n 分别表示交互比例的平均值和标准差。通过以下方式计算 $P(c_{ui} = 1)$:

$$P(c_{ui} = 1) = \text{sigmoid}\left(\frac{n_i - \bar{n}}{\sigma_n}\right). \quad (3.28)$$

其中 sigmoid 函数的作用是将输入转换为 $(0, 1)$ 范围内的概率。

3.2.5 最大化步骤

在最大化步骤中，固定隐变量 \hat{c}_{ui} ，然后学习模型参数 Θ 以最大化 $Q(\cdot)$ 函数，即

$$\arg \max_{\Theta} \sum_{(u,i,j) \in \mathcal{D}} \hat{c}_{ui} [\ln \sigma(\hat{x}_{ui} - \hat{x}_{uj}) - \lambda \|\Theta\|^2]. \quad (3.29)$$

采用广泛使用的随机梯度下降 (SGD) 技术来学习模型参数。对于每个训练三元组 (u, i, j) ，其梯度为：

$$\frac{\partial Q(\Theta, \Theta^{(t)})}{\partial \Theta} = \sum_{(u,i,j) \in \mathcal{D}} \hat{c}_{ui} [(1 - \sigma(\hat{x}_{ui} - \hat{x}_{uj})) \frac{\partial \hat{x}_{uij}}{\partial \Theta} - \lambda \Theta], \quad (3.30)$$

其中

$$\frac{\partial \hat{x}_{uij}}{\partial \Theta} = \frac{\partial (\hat{x}_{ui} - \hat{x}_{uj})}{\partial \Theta} = \begin{cases} \mathbf{h}_i - \mathbf{h}_j, & \text{if } \theta = \mathbf{w}_u \\ \mathbf{w}_u, & \text{if } \theta = \mathbf{h}_i \\ -\mathbf{w}_u, & \text{if } \theta = \mathbf{h}_j. \end{cases} \quad (3.31)$$

θ 表示 \mathbf{W} 或 \mathbf{H} 的特定行或者列。给定正则化参数 α ，以及正则化常数 λ ，模型参数的更新规则为

$$\Theta \leftarrow \Theta + \alpha \hat{c}_{ui} [(1 - \sigma(\hat{x}_{ui} - \hat{x}_{uj})) \frac{\partial \hat{x}_{uij}}{\partial \Theta} - \lambda \Theta]. \quad (3.32)$$

3.3 算法实现与时间复杂度分析

3.3.1 伪代码

从公式(3.32)可以看出，新的参数 c_{ui} 充当了动态学习率的角色，实质上重新加权了具有较高偏好水平的交互，并为学习算法过滤了具有较低偏好水平的噪声交互。算法的实现使用了 EM 框架，在 E 步固定模型参数求解隐变量，在 M 步固定隐变量学习模型参数。算法 1 给出了 BPRAC 学习算法的伪代码。

Algorithm 1 自适应去噪的成对学习排序算法伪码

Input: 交互集合 \mathcal{S} , 训练三元组集合 \mathcal{D} , 最大迭代轮数 T , 抽样的训练三元组数量 K

Output: 预测的评分矩阵 $\mathbf{X} \in \mathbb{R}^{M \times N}$

初始化用户物品表示 $\Theta = (\mathbf{W}, \mathbf{H})$, 其中 $\mathbf{w}_u \sim \mathcal{N}(0, \lambda\mathbf{I})$, $\mathbf{h}_i \sim \mathcal{N}(0, \lambda\mathbf{I})$;

for $t = 1; t \leq T; t++$ **do**

Expectation 步: 固定 Θ , 计算隐变量 \hat{c}_{ui} ;

for 每一个交互 $(u, i) \in \mathcal{S}$ **do**

计算预测评分 $\hat{x}_{ui} = \langle \mathbf{w}_u, \mathbf{h}_i \rangle$;

通过公式 (3.19)计算先验分布 $f(\hat{x}_{ui})$;

通过公式 (3.20)计算 $F(\hat{x}_{ui})$;

通过公式 (3.22)计算后验分布 $g(\hat{x}_{ui}; k = 2, n = 2)$;

通过公式 (3.27)计算 \hat{c}_{ui} ;

Maximization 步: 固定 \hat{c}_{ui} , 更新参数 Θ ;

for $k = 1; k \leq K; k++$ **do**

抽样一个三元组 $(u, i, j) \in \mathcal{D}$;

计算预测评分 \hat{x}_{ui} , \hat{x}_{uj} ;

通过公式 (3.32)执行梯度下降更新参数 Θ ;

Result: $\mathbf{X} = \mathbf{W} \times \mathbf{H}^\top$

3.3.2 时间复杂度分析

由于 E 步对隐变量的估计是标量计算, 因此时间复杂度是 $\mathcal{O}(|\mathcal{S}|)$, 其中 $|\mathcal{S}|$ 是训练集中交互总数。而 M 步主要是向量的梯度下降, 因此 M 步的时间复杂度是 $\mathcal{O}(|\mathcal{D}|d)$, 其中, $|\mathcal{D}|$ 是三元组的数量, d 是嵌入的维度。由于 $|\mathcal{S}| \leq |\mathcal{D}|$, 即交互数远小于训练三元组的数量, 因此自适应去噪的成对排序算法的时间复杂度主要源自于 M 步: 固定隐变量, 学习用户和物品的特征表示, 即执行一次最外层循环的时间复杂度为 $\mathcal{O}(|\mathcal{D}|d)$, 这也是标准的 BPR 的时间复杂度。由于最外层要执行 T 次循环, 总体的计算复杂度为 $\mathcal{O}(T|\mathcal{D}|d)$, 其中 T 是学习迭代的次数。

3.4 收敛性理论分析

在期望步骤中, 使用 $\mathbb{E}[c_{ui}] \propto \hat{c}_{ui}$ 的近似来计算 c_{ui} 的估计值。从公式(3.32)中可以直观地观察到, 由于 $\hat{c}_{ui} \geq 0$, \hat{c}_{ui} 的近似仅改变参数更新过程中的步长, 而不改变

参数更新的方向，也就是说这个近似只改变了梯度下降过程中通往（局部）最小值的步长，但并不会改变梯度更新的方向，因此并不会改变算法的收敛性。BPRAC 算法的收敛性的严格证明由如下引理给出。

引理 3.2：给定所有的成对比较 \succ_u ，记后验概率序列为 $P(\Theta^{(t)} | \succ_u)$ ，其中 $\Theta^{(t)} (t = 1, 2, \dots)$ 为第 t 训练轮次学到的用户物品表示。那么，后验概率 $P(\Theta^{(t)} | \succ_u)$ 收敛，且

$$\lim_{t \rightarrow \infty} P(\Theta^{(t)} | \succ_u) = \sup\{P(\Theta^{(t)} | \succ_u) | t \in \mathbb{N}\}. \quad (3.33)$$

证明：首先证明 $\mathbb{E}[c_{ui}] \propto \hat{c}_{ui}$ 的这个近似不改变后验概率序列的非递减性质，也就是说要证明

$$P(\Theta^{(t+1)} | \succ_u) \geq P(\Theta^{(t)} | \succ_u). \quad (3.34)$$

根据全概率公式，有

$$\begin{aligned} P(\Theta | \succ_u) &= \frac{P(\Theta, c | \succ_u)}{P(c | \Theta, \succ_u)} \\ &= \frac{P(\Theta | c, \succ_u) P(c | \succ_u)}{P(c | \Theta, \succ_u)}. \end{aligned} \quad (3.35)$$

因此

$$\log P(\Theta | \succ_u) = \log P(\Theta | c, \succ_u) - \log P(c | \Theta, \succ_u) + \log P(c | \succ_u). \quad (3.36)$$

定义：

$$\begin{aligned} Q(\Theta, \Theta^{(t)}) &= \mathbb{E}_{c_{ui}} [\log P(\Theta | c, \succ_u) | \succ_u, \Theta^{(t)}], \\ H(\Theta, \Theta^{(t)}) &= \mathbb{E}_{c_{ui}} [\log P(c | \Theta, \succ_u) | \succ_u, \Theta^{(t)}], \\ K(\Theta, \Theta^{(t)}) &= \mathbb{E}_{c_{ui}} [\log P(c | \succ_u) | \succ_u, \Theta^{(t)}]. \end{aligned} \quad (3.37)$$

于是

$$\log P(\Theta | \succ_u) \triangleq Q(\Theta, \Theta^{(t)}) - H(\Theta, \Theta^{(t)}) + K(\Theta, \Theta^{(t)}). \quad (3.38)$$

那么后验概率序列的差值为：

$$\begin{aligned} \log P(\Theta^{(t+1)} | \succ_u) - \log P(\Theta^{(t)} | \succ_u) &= [Q(\Theta^{(t+1)}, \Theta^{(t)}) - Q(\Theta^{(t)}, \Theta^{(t)})] \\ &\quad - [H(\Theta^{(t+1)}, \Theta^{(t)}) - H(\Theta^{(t)}, \Theta^{(t)})]. \end{aligned} \quad (3.39)$$

华 中 科 技 大 学 博 士 学 位 论 文

在期望步骤中，采用了 $\mathbb{E}[c_{ui}] \propto \hat{c}_{ui}$ 的近似方法。假设 $\hat{c}_{ui} = \text{const} \cdot \mathbb{E}[c_{ui}]$ ，其中 $\text{const} > 0$ 。根据最大化步骤，

$$\begin{aligned}\Theta^{(t+1)} &= \arg \max_{\Theta} \sum_{(u,i,j) \in D_S} \hat{c}_{ui} [\ln \sigma(\hat{x}_{ui} - \hat{x}_{uj}) - \lambda_\Theta \|\Theta\|^2] \\ &= \arg \max_{\Theta} \sum_{(u,i,j) \in D_S} \text{const} \cdot E(c_{ui}) [\ln \sigma(\hat{x}_{ui} - \hat{x}_{uj}) - \lambda_\Theta \|\Theta\|^2] \\ &= \arg \max_{\Theta} \text{const} \cdot Q(\Theta, \Theta^{(t)}).\end{aligned}\quad (3.40)$$

如上所示，使得 $\text{const} \cdot Q(\Theta, \Theta^{(t)})$ 最大化的 $\Theta^{(t+1)}$ 也同时使得 $Q(\Theta, \Theta^{(t)})$ 最大化，即 $Q(\Theta^{(t+1)}, \Theta^{(t)}) - Q(\Theta^{(t)}, \Theta^{(t)}) > 0$ 。因此，公式(3.39)的第一项是非负的。接下来，按照标准的 EM 算法^[113] 完成证明。公式(3.39)的第二项是：

$$\begin{aligned}& [H(\Theta^{(t+1)}, \Theta^{(t)}) - H(\Theta^{(t)}, \Theta^{(t)})] \\ &= \mathbb{E}_{c_{ui}} [\log P(c | \succ_u, \Theta^{(t+1)}) | \succ_u, \Theta^{(t)}] - \mathbb{E}_{c_{ui}} [\log P(c | \succ_u, \Theta^{(t)}) | \succ_u, \Theta^{(t)}] \\ &= \mathbb{E}_{c_{ui}} [\log \frac{P(c | \succ_u, \Theta^{(t+1)})}{P(c | \succ_u, \Theta^{(t)})} | \succ_u, \Theta^{(t)}] \\ &\leq \log \mathbb{E}_{c_{ui}} [\frac{P(c | \succ_u, \Theta^{(t+1)})}{P(c | \succ_u, \Theta^{(t)})} | \succ_u, \Theta^{(t)}] \\ &= \log \int \frac{P(c | \succ_u, \Theta^{(t+1)})}{P(c | \succ_u, \Theta^{(t)})} P(c | \succ_u, \Theta^{(t)}) dc \\ &= 0.\end{aligned}\quad (3.41)$$

根据公式 ((3.40)) 和 ((3.41))，得出结论公式 ((3.39)) 是非负的，也就是说，后验概率序列 $P(\Theta^{(t)} | \succ_u)$ 是非递减的。

其次， $P(\Theta^{(t)} | \succ_u)$ 是观察到的成对比较的后验概率，因此

$$P(\Theta^{(t)} | \succ_u) \leq 1. \quad (3.42)$$

由于 $P(\Theta^{(t)} | \succ_u)$ 是非递减的，并且 1 是其上界之一，根据单调有界收敛定理，对于 $t \in \mathbb{N}$ ，后验概率序列存在极限，并且

$$\lim_{t \rightarrow \infty} P(\Theta^{(t)} | \succ_u) = \sup \{P(\Theta^{(t)} | \succ_u) | t \in \mathbb{N}\}. \quad (3.43)$$

证毕。 \square

表 3.1 数据集统计信息

数据集	用户数	物品数	训练集交互数	测试集交互数	密度
ML-100k	943	1,682	89,372	10,628	6.30%
ML-1M	6,040	3,952	887,007	113,202	4.19%
Yahoo!-R3	5,400	1,000	129,180	12,520	2.62%

3.5 试验评估

3.5.1 实验设置

数据集: 本章的在三个公开数据集上进行实验, 包括 MovieLens-100k、MovieLens-1M 和 Yahoo!-R3^[16]。它们包含用户对物品的评级, 根据离散评分系统进行评定, 最满意为五分, 最不满意为一分。在从隐式反馈中进行个性化排序的研究中, 一种常用的数据预处理技术是将所有已评级的物品转换为隐式反馈^[1,14,18]。在模型训练时, 只能观测到交互, 但不包括任何评分细节。

注意到四分或五分的评级往往能更好地反映用户对互动物品的真实偏好。因此, 构建了仅包含这些物品的“干净”测试集^[114], 即预测用户是否真正喜欢某个物品, 而不是他是否有可能给该物品评分。对于 MovieLens-100K 和 MovieLens-1M 数据集, 随机选择 50% 的物品, 每个物品都有五分评级作为测试数据。对于 Yahoo!-R3 数据集, 测试集中具有五分评级的物品很少。为了避免测试集中物品过少, 还选择了被评为 3 星的歌曲^[114], 因为这样的歌曲也能反映出一定程度的用户偏好。表 3.1 总结了数据集的统计信息。

对比方法: 本章提出的自适应去噪的成对排序算法 (Bayesian Personalized Ranking with Autonomous Credence, BPRAC) 旨在从含噪成对比较中学习排序, 因此同类对比方法主要为面向隐式反馈数据的成对学习算法。

- BPR^[1]: 作为先驱工作, 它在本文中已经被很好地介绍过。
- GBPR^[13]: GBPR 为用户 u 引入了群体偏好的概念 $\mathcal{G}(i)$, 其中 $\mathcal{G}(i)$ 还包括与物品 i 有交互的其他用户。群体偏好的引入可以有效避免正例噪声问题。
- MPR^[14]: MPR 构建了更细粒度的偏好比较, 能够一定程度上缓解噪声问题:

$$r_{ui} - r_{uj_2} \geq r_{uj_1} - r_{uj_2} \geq r_{ui_1} - r_{ui_2}, \quad i, i_1, i_2 \in \mathcal{I}_u^+, \quad j_1, j_2 \in \mathcal{I}_u^-,$$
 其中 j_1 是未见但可能喜欢的物品, j_2 是负面物品, 从最不受欢迎的物品中选择。
- DenoiseRec^[114]: DenoiseRec 率先在推荐系统中引入小损失技巧, 其核心思想是

表 3.2 隐变量的估计质量

Dataset	MovieLens-100k	MovieLens-1M	Yahoo!-R3
AUC	0.6524	0.6618	0.5885

损失值较大的样本可能是噪声。具体而言，对于那些大于 τ 的损失值的样本，DenoiseRec 认为很有可能是噪声交互，并使用动态阈值函数将损失值截断为 0，或者使用较小的权重进行重新加权。

参数设置：模型的超参数 d 和 λ 在对比算法中已经进行了详细讨论^[1,13,14]。遵循它们的讨论，在三个数据集中将 d 设置为 27，并在 MovieLens-1K 和 MovieLens-1M 数据集中将 λ 设置为 0.6，在 Yahoo!-R3 数据集中将 λ 设置为 0.2。将迭代次数 $T = 10$ ，学习率 $\alpha = 0.05$ ，采样的三元组数量 $K = 1000 \times$ 用户数量，对于所有三个数据集保持一致。对比算法中的其他超参数是根据其收敛 AUC 指标的最佳性能进行选择的。

评估指标：按照 BPR^[1] 的做法，采用广泛使用的 AUC 作为性能指标之一。除了用于成对比较的 AUC 性能指标外，还采用常用的用于 Top-K 个性化推荐的评估指标，包括 Precision、Recall、F1、NDCG（归一化折现累积增益）和 MAP（平均准确率）。由于评估指标的广泛使用，受限于篇幅，详细定义参考^[115]。

3.5.2 实验结果

\hat{c}_{ui} 估计的质量：估计 \hat{c}_{ui} 的任务可以直观理解为预测隐式反馈数据集中每个交互对应的偏好值大小。 \hat{c}_{ui} 的值越高，交互被认为越可信，偏好值也就越高。在训练数据集中，由于有评分，就可以根据评分来判定 \hat{c}_{ui} 的预测好坏。特别地，把训练集中评分为 1 的交互设置为不可信交互，那么就可以对 \hat{c}_{ui} 估计进行评估。图 3.4 绘制了通过设置不同估计阈值的所有交互的 \hat{c}_{ui} 的 ROC（接收者操作特性）曲线，并且表 3.2 计算了三个数据集对应的 AUC。首先承认， \hat{c}_{ui} 的 AUC 性能并不是很令人满意，但它们仍然比简单猜测抛硬币要好，显著大于 50%。这意味着，BPRAC 算法总体上能够为偏好较高的交互输出一个较大的 \hat{c}_{ui} 估计值，为偏好较低的交互输出一个较小的 \hat{c}_{ui} 估计值。在仅具有二元交互矩阵的有限信息下，这样的增益主要来自于后验分布的收窄，从而减小了不确定性，如公式 (3.26) 所描述。

个性化推荐结果性能评估：表 3.3 比较了四种成对学习排序模型的推荐性能。BPRAC 模型在所有评估指标中表现最好。DenoiseRec 取得了第二好的性能，表明从

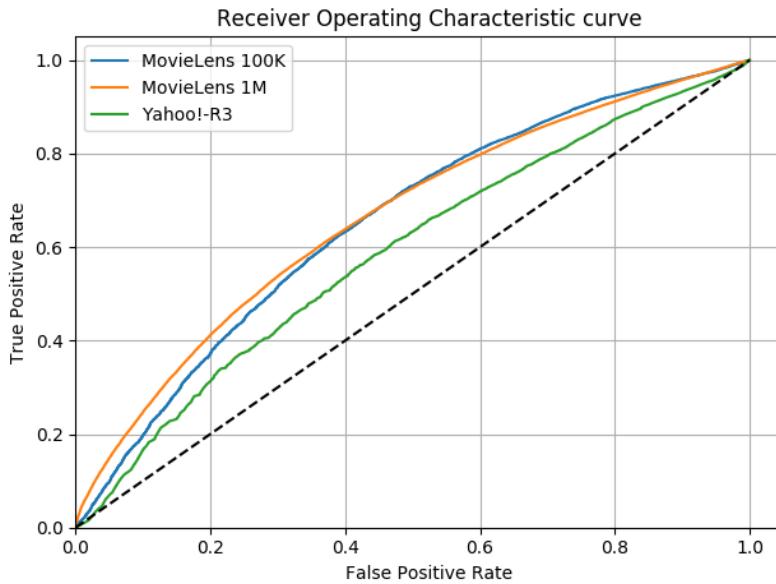


图 3.4 隐变量估计的 ROC 曲线

隐式反馈数据中去噪是必要的。然而，与 BPRAC 相比，DenoiseRec 收敛到一个较小的 AUC 值，这意味着 BPRAC 中的参数学习比使用阈值经验确定交互是否为噪声数据更有效。此外，其余的三个对比算法都没有在隐式反馈数据中的去噪机制，每个交互都被认为是体现用户偏好的交互，这显然是不符合实际情况的。BPRAC 的性能改进一方面表明不可信交互的存在对成对学习模型产生了不利影响；另一方面，验证了基于 EM 的学习模型的有效性。

对噪声数据的敏感性：通过在训练集中随机生成噪声数据来研究 BPRAC 对噪声数据的敏感性。对比方法选择 BPR 和 DenoiseRec。生成包含噪声的训练集时，不同的学习算法使用相同的含有噪声的训练集训练。如图 3.5 所示，噪声数据的增加导致三种学习算法的个性化推荐性能都下降，表明不可信的交互存在对学习模型产生不利影响；另一方面，由于 BPRAC 和 DenoiseRec 包含去噪机制，它们相对于 BPR 更具鲁棒性。此外，DenoiseRec 的性能下降略多于 BPRAC，这意味着通过自适应地学习置信度，以去除噪声，比经验地通过阈值判断数据是否为噪声数据更有效。

AUC 收敛性：图 3.6 比较了五种算法的收敛性，度量指标为排序列表的 AUC 指标，这是因为优化目标函数与 AUC 指标是可以类比的^[1]。首先观察到，所有算法的 AUC 在足够的训练迭代后首先增加，然后收敛。这个试验结果也印证了引理 3.2 的分析，如公式 (3.33) 所示。此外注意到，与对比算法相比，本章提出的 BPRAC 收

表 3.3 Top-k 推荐性能比较

数据集	方法	Top-5					Top-10				
		Precision	Recall	F1	NDCG	MAP	Precision	Recall	F1	NDCG	MAP
MovieLens100K	BPR	0.2888	0.1897	0.1818	0.3401	0.4893	0.2379	0.2923	0.2075	0.3422	0.4663
	GBPR	0.2976	0.1944	0.1872	0.3504	0.4967	0.2458	0.3031	0.2147	0.3538	0.4780
	MPR	0.1224	0.0741	0.0754	0.1333	0.2278	0.1053	0.1331	0.0935	0.1412	0.2295
	DenoiseRec	0.3042	0.1890	0.1883	0.3528	0.5122	0.2541	0.3027	0.2192	0.3553	0.4955
	BPRAC	0.3265	0.2158	0.2055	0.3879	0.5393	0.2665	0.3216	0.2305	0.3859	0.5127
MovieLens1M	BPR	0.2940	0.1086	0.1323	0.3148	0.4645	0.2570	0.1825	0.1741	0.3042	0.4517
	GBPR	0.3119	0.1265	0.1497	0.3376	0.5004	0.2655	0.2027	0.1874	0.3235	0.4797
	MPR	0.1125	0.0410	0.0494	0.1212	0.2226	0.1036	0.0713	0.0681	0.1207	0.2278
	DenoiseRec	0.3205	0.1174	0.1434	0.3432	0.4996	0.2737	0.1901	0.1831	0.3257	0.4783
	BPRAC	0.3472	0.1282	0.1572	0.3721	0.5297	0.2992	0.2110	0.2026	0.3559	0.5044
Yahoo!-R3	BPR	0.0861	0.1007	0.0840	0.1099	0.1890	0.0685	0.1566	0.0875	0.1284	0.1962
	GBPR	0.1044	0.1250	0.1037	0.1343	0.2292	0.0817	0.1943	0.1063	0.1576	0.2351
	MPR	0.0694	0.0642	0.0511	0.0744	0.1353	0.0551	0.1095	0.0527	0.0875	0.1503
	DenoiseRec	0.1058	0.1257	0.1042	0.1349	0.2247	0.0841	0.1993	0.1086	0.1596	0.2326
	BPRAC	0.1117	0.1308	0.1096	0.1448	0.2465	0.0857	0.2034	0.1109	0.1675	0.2510

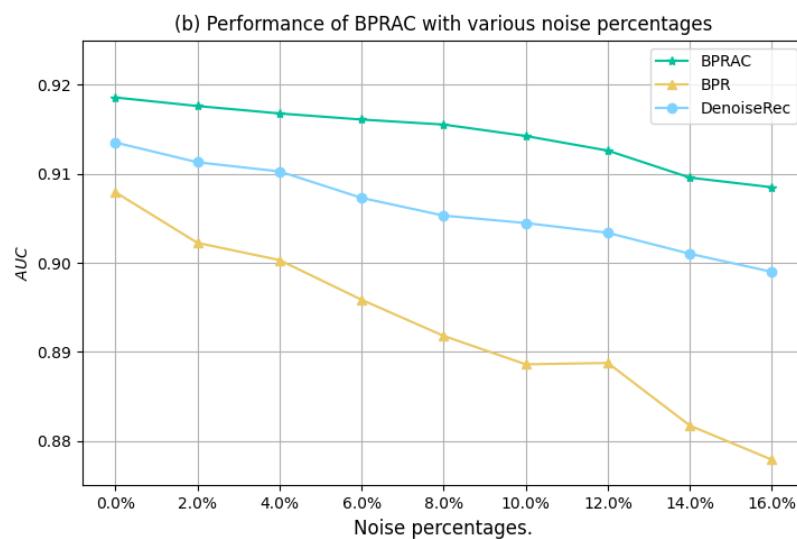


图 3.5 不同噪声率对学习算法性能的影响

致到一个更高的 AUC 指标，代价是更慢的收敛速率从而需要更多的训练轮次，通常需要比 BPR 多进行 2-3 倍更新，这是由于引入了新的隐藏变量 c_{ui} 。这个代价是值得的，因为 BPRAC 提供了一种从噪声数据中学习排序的解决方案，它放宽了每个交互都表达真实偏好的强假设，取得了更好的性能。

3.6 本章小结

本章聚焦于推荐系统的伪正例问题，研究了从含噪成对比较中学习排序的问题。由于隐式反馈是不完全数据，其中的伪正例构建的噪声成对比较，会对学习算法产生不利的影响。对于每个交互数据，本章引入了一个隐变量指示其可信度，并把从不完全数据中学习排序的问题形式化为一个含有隐变量的最大后验估计问题。针对这一问题，本章提出了自适应去噪的成对排序算法（BPRAC），端到端地估计隐变量并学习用户和物品表示。BPRAC 算法在 EM 框架下求解：在 E 步固定模型参数估计隐变量，在 M 步固定隐变量学习模型参数。对真实的推荐数据集进行的实验 BPRAC 算法相对于其他对比方法的优越性。

值得注意的是，本章的解决方案仅限于基于矩阵分解的模型，探索如何将所提出的学习算法推广到更先进的基于神经网络的学习方法，如 LightGCN^[5] 将是一个重要的问题。对于这些基于神经网络的学习方法，计算先验分布 $f(\hat{x}_{ui})$ 是复杂的。此外，伪负例也是隐式反馈数据所面临的一个突出问题。本章的研究将为后续章节中探索更复杂模型和更一般的伪负例场景的解决方案提供启示。

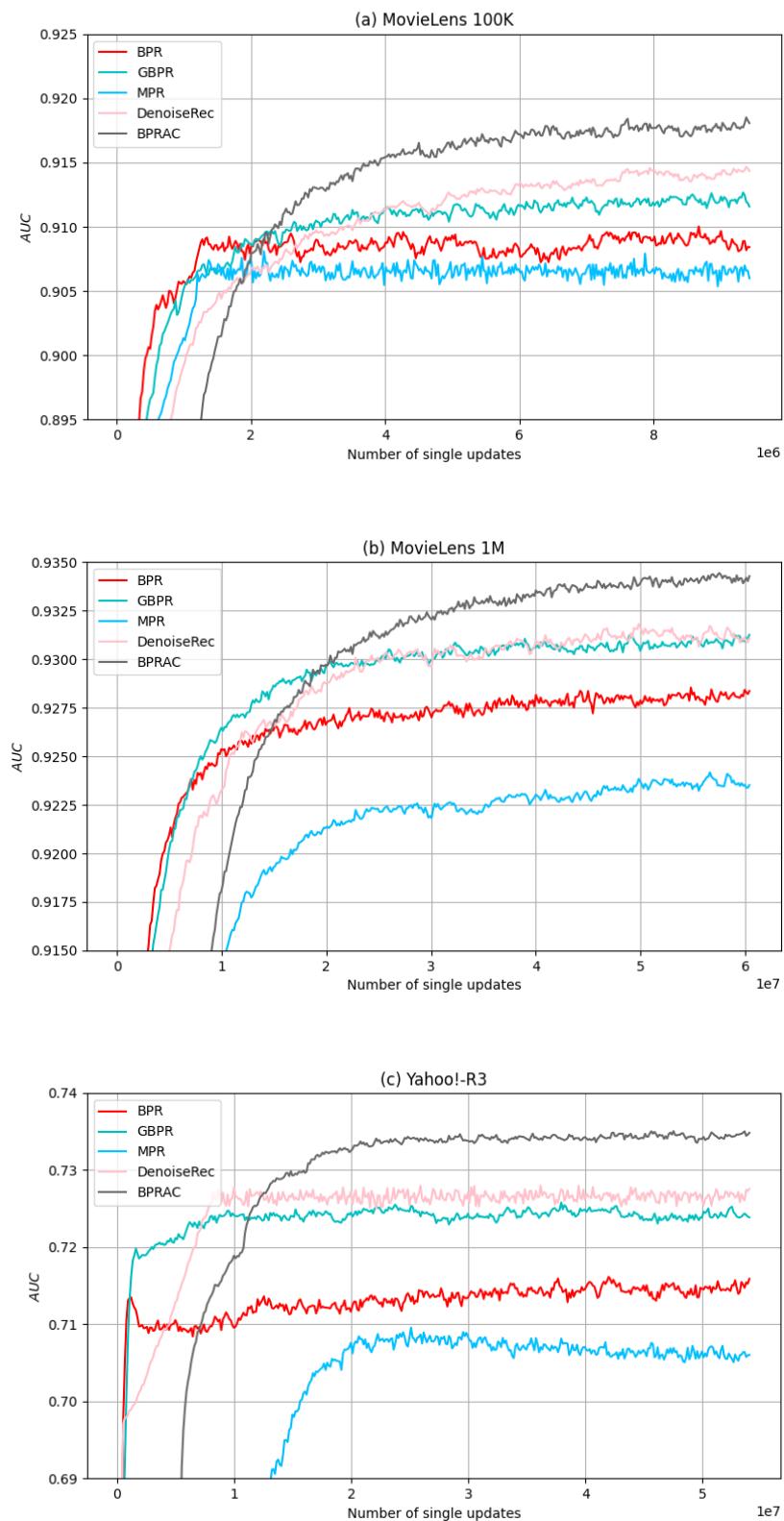


图 3.6 不同学习算法的收敛速度比较

4 负例采样算法研究

本章聚焦推荐系统中的伪负例问题。隐式反馈数据面临的一个突出问题是负例未标注。通常，未交互的物品被当作负样本以训练模型。然而，其中存在没曝光给用户，但是用户可能喜欢的物品，从而导致了伪负例问题。这一问题在自监督学习的诸多领域具有普遍性，例如在自然语言处理（NLP）中，正样本可以从上下文中获得，而负样本则是从未标注的词库中随机抽样得到的。在计算机视觉（CV）中，正样本通过数据增强获得，而负样本则从未标记的图像中抽样得到。这样的数据集也被称为 Positive-Unlabeled (PU) 数据集。如何从未标记的数据中采样高质量的负实例，即负采样，对于训练基于对比学习的模型非常重要。本章聚焦于推荐系统中负采样问题的研究，通过设计最优负采样准则，采样信息量大且用户不喜欢的硬负样本（hard negative），以提升推荐精度。

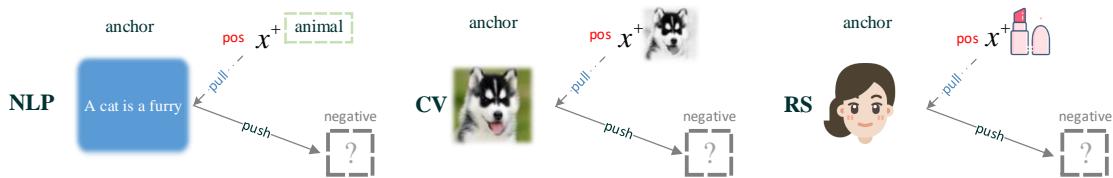


图 4.1 Positive-Unlabeled(PU) 问题示意图

4.1 引言

负采样源自 Positive-Unlabeled (PU) 问题^[116,117]。一个典型的 PU 数据集，通常只有部分正例被标注，其余样本未标注，而这些未标注的样本可能属于正类或负类。负采样，即确定从 PU 数据集中采样负例的策略，以便有效地训练下游任务模型。负采样在许多任务中都有广泛应用，例如自然语言处理（NLP）^[53,118]，计算机视觉（CV）^[50,51]，以及推荐系统（RS）^[3,45,46]。本章聚焦于推荐系统中的负采样。对于一个特定的用户，他未交互的物品，称为他的负例。里面包含用户确实不喜欢的物品，称为真负例；也包含用户喜欢的物品，称之为伪负例（即正例）。负采样的困境在于，未标注的样本中含有用户潜在喜欢的伪负例，一旦这类用户感兴趣的样本被

采样当作负例用于训练模型，会使得模型误判用户的兴趣边界。这激发了负采样用于推荐的问题，即如何有效地采样负实例来训练推荐模型。许多研究表明，负采样对于提高推荐性能非常重要^[3,45,46,59–62]。

负采样算法在推荐系统中被广泛研究。根据抽样分布是否随模型状态的变化，可以将现有的负采样算法分为以下两类：静态负采样^[1,5,6,13,14,52,53]。这类算法根据固定的采样分布，如均匀采样，依物品流行度的采样等，这类抽样分布所参照的信息是静态的，不随模型训练的参数变化而变化。另外一类是动态负采样^[45,46,63,65]。这类负采样算法的抽样分布会根据模型训练不断调整，偏好于采样在嵌入空间中与正例表示更相似的负例，即难以区分的困难样本（hard negative sample），例如采样分数更高或排名更高的实例^[45–47]。静态负采样可以使用某些辅助信息，免于采样到用户感兴趣的伪负例。但是由于其抽样分布是静态的，往往只会采样容易样本，不利于模型性能的提升；动态负采样可以根据模型的训练针对困难样本进行采样，更容易受到负采样问题的困扰^[3,50,51]。

本章对推荐系统的负采样研究做出了以下三个方面的贡献：(1) 根据经验分布函数的渐近性质，定义了一个与模型无关的负信号测度，用于度量未标注样本是真负例的后验概率。特别地，它较好地结合了现有基于先验信息（与模型无关）以及基于样本信息（与模型相关）的两种提取负信号的范式。(2) 本章还提出了最优采样准则，用于采样高质量的负例。这是一个经验采样风险最小化的理论最优采样准则。实验证明了所提出的负采样算法在采样质量和推荐性能方面的有效性。

4.2 负采样分析

对于一个由 M 个用户和 N 个物品组成用户-物品交互矩阵 $\mathbf{X} = [x_{ui}] \in \mathbb{R}^{M \times N}$ ，元素 $x_{ui} = 1$ 表示用户 u 与物品 i 有交互；否则， $x_{ui} = 0$ 。为了训练推荐模型 \mathcal{M} ，我们考虑如下广泛使用的 BPR 优化目标：

$$\mathcal{L} = \sum_{(u,i,j)} \ln \sigma(\hat{x}_{ui} - \hat{x}_{uj}), \quad (4.1)$$

对于用户 u ，其中 \hat{x}_{ui} 和 \hat{x}_{uj} 分别是他已交互物品 i 和未交互物品 j 的预测分数，由用户物品表示 Θ 参数化： $\hat{x}_{ui} = \text{sim}(\mathbf{w}_u, \mathbf{h}_i)$ ，其中 sim 是相似度函数，例如余弦相似度或点积函数。 (u, i, j) 为训练三元组。训练三元组 (u, i, j) 的构建如下：对于用户 u

和他的一一个正例 $i \in \mathcal{I}_u^+$, 负例从未标注样本中随机采样一个 $j \in \mathcal{I}_u^-$, 即进行负采样。存在着两种情况, 即用户 u 实际上喜欢 j , 未交互仅仅是因为之前没有看到过它, 也就是说, 物品 j 对于用户 u 来说是一个伪负例。另一方面, 如果用户 u 确实不喜欢物品 j , 则该物品被称为真负例。

通常使用随机梯度下降 (SGD) 来迭代优化每个训练三元组 (u, i, j) 的损失函数。对于采样的实例 $j \in \mathcal{I}_u^-$, 如果它是用户 u 的真负例, 则相对于其预测得分 \hat{x}_{uj} 的损失梯度计算如下

$$\frac{\partial \mathcal{L}}{\partial \hat{x}_{uj}} = [1 - \sigma(\hat{x}_{ui} - \hat{x}_{uj})](-1), \quad (4.2)$$

其中 $\sigma(\cdot)$ 是 sigmoid 函数。然而, 如果采样的实例 j 实际上是一个伪负例, 根据微分的链式法则, 正确的梯度更新应该是相反, 即公式 (4.2) 的最后一项 (-1) 改为 $(+1)$ 。由此可见, 对于一个未标注样本, 由于其标签不同, 梯度方向也截然相反。因此可以把损失函数相对于未标注样本本地损失梯度写为:

$$\frac{\partial \mathcal{L}_{loss}}{\partial \hat{x}_{uj}} = [1 - \sigma(\hat{x}_{ui} - \hat{x}_{uj})] \cdot \text{sgn}(j), \quad (4.3)$$

其中, 如果 j 是用户 u 的真负例, 则 $\text{sgn}(j) = -1$; 如果 j 是用户 u 的误判负例, 则 $\text{sgn}(j) = 1$ 。损失函数的梯度公式 (4.3) 可以分解为两部分, 即梯度幅度和梯度方向。这激发了我们对负采样的分析, 即什么是高质量的负例: 如果在训练三元组 (u, i, j) 中采样的实例 j 同时具有信息量和无偏性, 那么它被称为高质量负例。

- **信息量:** 未标注样本 j 的信息量定义为该样本对应的损失函数梯度幅度的大小, 即

$$\text{info}(j) = [1 - \sigma(\hat{x}_{ui} - \hat{x}_{uj})]. \quad (4.4)$$

- **无偏性:** 未标注样本 j 的信息量定义为该样本是真负例的后验概率大小, 即

$$\text{unbias}(j) = P(\text{sgn}(j) = -1). \quad (4.5)$$

信息量与未标注样本 j 对于推荐模型参数更新的帮助程度相关。给定正例的预测得分 \hat{x}_{ui} , 如果未标注样本 \hat{x}_{uj} 的值过小, 则 $\sigma(\hat{x}_{ui} - \hat{x}_{uj}) \rightarrow 1$ 且 $\text{info}(j) \rightarrow 0$, 即梯度消失, 无法从未标注样本 j 上学到太多信息。这是困难负采样 (hard negative sampling) 的一个重要动机。

无偏性实际上定义为未标注样本 j 是真负例的概率。对于负采样而言，正例已经给定，目标是采样真负例。在训练时，把采样到的未标注样本当作负例，即直接设置 $\text{sgn}(j) = -1$ 。当存在采样偏差时，即采样到正例，将导致推荐模型误判用户的兴趣边界。如何从未标记的样本中识别出真负例示例是必须解决的负采样关键研究问题。

采样偏差的不良影响不仅限于 BPR 损失，上述分析可以同样推广到其他一些基于对比的损失函数如 InfoNCE，也可能因错误的负采样而受到影响：当采样一个伪负例时，损失函数对于该采样的样本梯度方向错误。这个伪负例的特征表示将被推离锚点 u ，而实际上它应该在嵌入空间中靠近锚点。

4.3 贝叶斯最优负采样算法

4.3.1 类条件概率密度

由于所有未标注样本的预测得分都是使用相同的编码器和相同的得分函数预测的，因此可以假设预测得分 \hat{x} 是独立同分布的，其概率密度函数记为抽象函数 $f(\hat{x})$ ，相应的累积分布函数为 $F(\hat{x}) = \int_{-\infty}^{\hat{x}} f(t)dt$ 。

考虑未标注样本中的两个总体：真负例和伪负例。根据成对学习的优化目标，是针对锚点和正例的相似度分数高于负例的相似度分数进行优化。这表明以下预测得分的次序关系可能在一般情况下成立

$$X_{tn} \leq X_{fn}, \quad (4.6)$$

其中， X_{tn} 和 X_{fn} 分别为真负例和伪负例（即未来测试数据中的正例）的预测得分的随机变量。需要注意的是，这也是对比学习方法^[7,24,25]的优化目标，其本质上鼓励学到的正例特征表示与“锚点”数据点相似，同时将负例特征与“锚点”数据在嵌入空间中推远，使其不相似^[26,32,119]。

真负例和伪负例两个总体预测分数的这种序关系，可以使用次序统计量的技术写出真负例和伪负例这两个总体相似度分数分布：

$$g(\hat{x}_{tn}) = 2f(\hat{x}_{tn})[1 - F(\hat{x}_{tn})], \quad (4.7)$$

$$h(\hat{x}_{fn}) = 2F(\hat{x}_{fn})f(\hat{x}_{fn}). \quad (4.8)$$

4.3.1.1 经验分布

为了验证公式(4.6)的次序关系，在模型训练过程中记录真负例总体(TN)和伪负例总体(FN)相似度分数的经验分布。其中，伪负例总体是测试集中的物品，在训练过程中无法观测；真负例总体是未交互物品中，除去测试集之外的其余物品。在MovieLens-100K数据集上，采用经典的矩阵分解推荐模型，并采用随机负采样在训练集上训练模型，并记录每个训练周期的预测得分。通过统计真负例和误判负例的预测得分，在图4.2中报告了它们在模型训练中的分布密度。

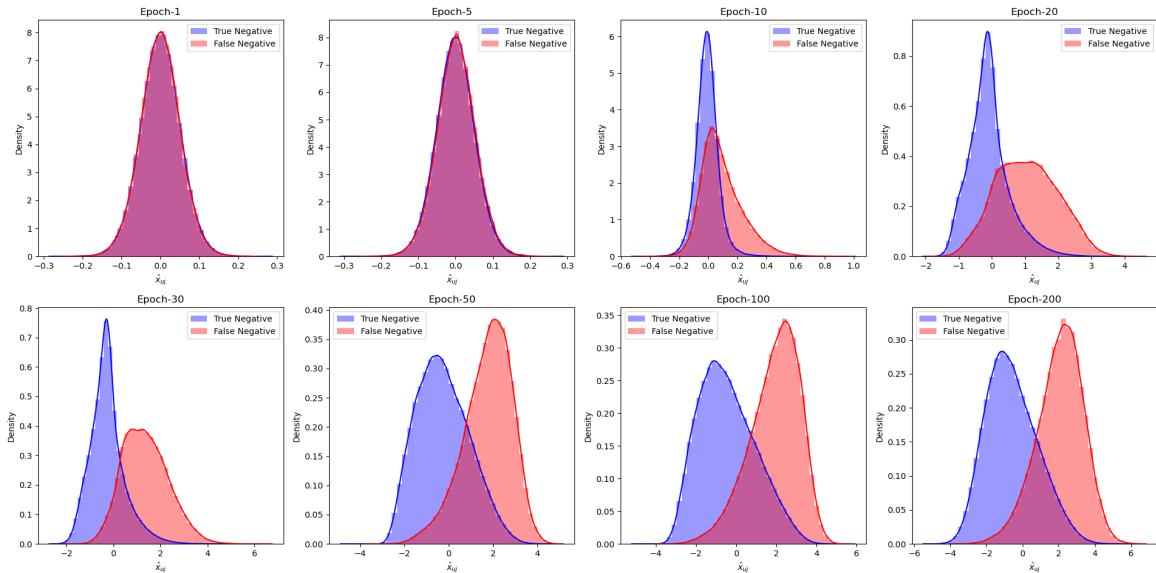


图4.2 不同训练时点的相似度分数的经验分布

图4.2提供了两个发现：

- 负例的预测得分越高，它是伪负例的概率密度就越高，而是真负例的概率密度就越低；
- 随着训练的进行，两个分布之间的区别逐渐变得更加清晰：与真负例相比，伪负例的分布集中在更高的得分上。这表明，相对于真负例而言，推荐模型总体上能够给伪负例更高的评分。

4.3.1.2 理论分布

图4.3展示了基于不同类型的分布 $f(x)$ 时（高斯分布 $x \sim \mathcal{N}(\mu, \sigma)$ ，学生分布 $x \sim t(n)$ 和伽玛分布 $x \sim Ga(\alpha, \lambda)$ ），公式(4.7)所对应的真负例和伪负例的分数分布

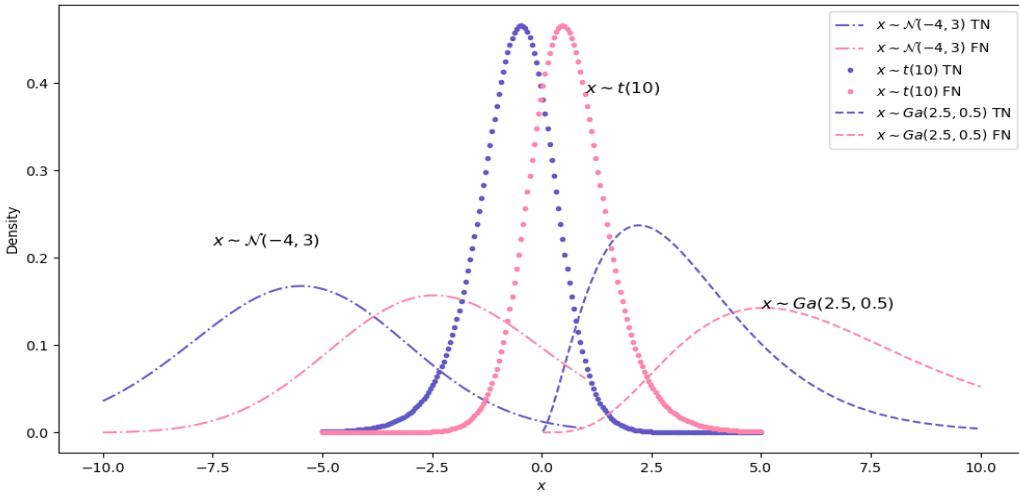


图 4.3 相似度分数的理论分布

形态。在训练过程中，使用真实数据集绘制的经验分布图 4.2，逐渐展示出了与图 4.3 中描述的理论分布出现了相同结构。

目前为止，我们还不知道 $f(x)$ 和 $F(x)$ 的显式表达式。不同的得分函数和推荐模型会产生不同的未标注样本分数密度表达 $f(\cdot)$ ，但这种分离的结构足以让我们对真负例和伪负例进行分类。此外，经验分布函数 $\hat{F}(x) = \frac{1}{n} \sum_j I_{|X_j \leq \hat{x}_l|}$ 的计算很容易实现。根据 Glivenko 定理^[100]，经验分布函数一致收敛于分布函数，使得可以通过计算经验分布函数，以实现对抽象函数 $F(\cdot)$ 的近似。这一有力工具，使得本章所提出的负采样方法不再局限于简单的矩阵分解模型，可以向任何深度神经网络模型推广。

4.3.2 贝叶斯分类

对于一个未交互的物品 l 以及其对应的预测得分为 \hat{x}_l ，使用贝叶斯公式计算物品 l 为真负例的后验概率：

$$\begin{aligned} P(tn|\hat{x}_l) &\propto P(\hat{x}_l|tn)P_{tn}(l) \\ &= 2f(\hat{x}_l)[1 - F(\hat{x}_l)]P_{tn}(l), \end{aligned} \quad (4.9)$$

其中， $P(\hat{x}_l|tn)$ 是真负例的类条件密度，由 $g(\hat{x})$ 给出； $P_{tn}(l) = 1 - P_{fn}(l)$ 是物品 l 为真负例的先验概率。 $f(\hat{x}_l)$ 是所有未标注样本的得分分布， $F(\hat{x}_l) = \int_{-\infty}^{\hat{x}_l} f(t)dt$ 是

相应的累积分布函数。同样地，未标注样本 l 是伪负例的后验概率为

$$\begin{aligned} P(f_n|\hat{x}_l) &\propto P(\hat{x}_l|f_n)P_{f_n}(l) \\ &= 2F(\hat{x}_l)f(\hat{x}_l)P_{f_n}(l) \end{aligned} \quad (4.10)$$

贝叶斯分类器可以通过最大化后验概率来得到：

$$\arg \max_{c \in \{f_n, t_n\}} P(c|\hat{x}_l). \quad (4.11)$$

直接计算公式 (4.9) 和公式 (4.10) 中的得分密度函数 $f(\cdot)$ 并不实际。然而，计算经验分布函数却很容易实现。因此，使用贝叶斯公式的分式形式定义无偏性，以消除密度函数 $f(\cdot)$ ：

$$\text{unbias}(l) \triangleq P(t_n|\hat{x}_l) = \frac{P(t_n, \hat{x}_l)}{P(\hat{x}_l)} \quad (4.12)$$

$$= \frac{P(\hat{x}_l|t_n)P_{t_n}(l)}{P(\hat{x}_l|t_n)P_{t_n}(l) + P(\hat{x}_l|f_n)P_{f_n}(l)} \quad (4.13)$$

$$\begin{aligned} &= \frac{f(\hat{x}_l)[1 - F(\hat{x}_l)]P_{t_n}(l)}{f(\hat{x}_l)[1 - F(\hat{x}_l)]P_{t_n}(l) + F(\hat{x}_l)f(\hat{x}_l)P_{f_n}(l)} \\ &= \frac{[1 - F(\hat{x}_l)][1 - P_{f_n}(l)]}{1 - F(\hat{x}_l) - P_{f_n}(l) + 2F(\hat{x}_l)P_{f_n}(l)}. \end{aligned} \quad (4.14)$$

根据 Glivenko 定理^[100]，可以使用 $\hat{F}(\hat{x}_l)$ 来近似 $F(\hat{x})$ ，即 $\hat{x} \leq \hat{x}_l$ 的比例：

$$F(x) \simeq \frac{1}{n} \sum_j I_{|X_j \leq \hat{x}_l|} \quad (4.15)$$

$P_{f_n}(l)$ 是物品 l 为伪负例的先验概率。假设物品 l 被交互的次数 $pop_l \sim B(N, P_{f_n}(l))$ ，其中 N 是训练集中的总交互次数。那么，

$$P_{f_n}(l) = \frac{pop_l}{N}. \quad (4.16)$$

无偏性的定义是物品 l 为真负例的后验概率。由于公式 (4.14) 的分数表达式，未标注样本分数的密度函数 $f(\hat{x}_l)$ 被消除掉。公式 (4.14) 意味着未交互物品 l 的负信号形式上由以下因素决定：(i) 推荐模型输出的排序位置，用 $F(\hat{x}_l)$ 表示，其中排序位置和 $F(\hat{x}_l)$ 具有一一映射关系。较大的 $F(\hat{x}_l)$ ，即靠前的排序位置，表明推荐模型在很大程度上将 l 分类为用户感兴趣的伪负例（正例），即 $F(\hat{x}_l)$ 分配了一个在 $[0, 1]$ 范围内的 l 是伪负例（正例）的概率预测。这解释了为什么那些过采样得分较高或

排名较高的困难负例的算法^[45,46]更容易采样伪负例的问题。(ii) 先验信息，表示为先验概率($P_{tn}(l)$ 或 $P_{fn}(l)$)。

注意到，公式(4.14)的度量标准统一了当前关于提取负信号的两种主要范式：(i) 建模先验信息，如曝光数据^[4]、知识图谱实体^[63]、社交网络中的连接^[68,69]等；(ii) 建依赖模型的样本信息，如预测得分^[45]、排名位置^[46]、得分方差^[3]等。前者可以融入领域知识，但负信号与模型状态无关；后者利用样本信息 \hat{x}_l ，但忽略了先验信息。 $\text{unbias}(\cdot)$ 测度的优势在于其对应的后验概率的理论解释：它结合了先验信息（与模型无关）和样本信息 \hat{x}_l （与模型相关）。本章只是为了简化，采用了一种简单的方法来建模先验概率，但不局限于此， $P_{tn}(l)$ 为使用其他附加信息和领域知识来建模先验概率提供了灵活的接口。

4.3.3 最优采样准则

公式(4.1)的成对损失 \mathcal{L} 可以类比于AUC指标^[1]。它用可微的损失函数 $\ln \sigma(\cdot)$ 替代了AUC指标中的不可微的0-1损失，这在优化AUC时是一种常见做法^[1,120]，因此关心 \mathcal{L} 的大小也是关心AUC指标的大小。

先在给定正例对 (u, i) ，采样到的样本 l 直接被赋予一个负向的梯度，导致预测得分 \hat{x}_{ul} 的减少，这会对 \mathcal{L} 产生两种影响：(i) 当 l 是伪负例时， \mathcal{L} 会减少，表示为 $\Delta\mathcal{L}_{fn}(l|i)$ ；(ii) 当 l 是真负例时， \mathcal{L} 会增加，表示为 $\Delta\mathcal{L}_{tn}(l|i)$ 。

定义4.1(条件采样风险): 对于给定的正例对 (u, i) ，定义采样未标注样本 l 的条件采样风险为，采样损失 $\Delta\mathcal{L}(l|i)$ 相对于 l 的标签 c 的期望值：

$$\begin{aligned} R(l|i) &= \mathbb{E}_{l \sim P(c|l)} \Delta\mathcal{L}(l|i) \\ &= [1 - P(tn|l)] \cdot \Delta\mathcal{L}_{fn}(l|i) + P(tn|l) \cdot \Delta\mathcal{L}_{tn}(l|i), \end{aligned} \quad (4.17)$$

其中 $P(tn|l)$ 是 $l \in tn$ 的后验概率，可以使用 $\text{unbias}(l)$ 计算， $1 - P(tn|l)$ 是 $l \in fn$ 的后验概率。条件采样风险的含义是，给定正例*i*，采样*l*所带来的AUC的期望减小值。

定义4.2(经验采样风险): 采样器*h*的经验采样风险为，条件采样风险 $R(l|i)$ 对所有正例分布的期望值：

$$R(h) = \mathbb{E}_{i \sim P(i)} R(l|i). \quad (4.18)$$

华 中 科 技 大 学 博 士 学 位 论 文

经验采样风险的含义为，对于所有正例，采样器所带来的 AUC 的期望减小值。

任务是找到一个采样策略 $h : \mathcal{I}_u^- \rightarrow l$ ，以最小化经验采样风险，从而实现最大化 AUC。如下定理给出了经验采样风险最小化意义的最优采样准则：

定理 4.1 (最优采样准则): 若条件采样风险 $R(l|i)$ 相互独立，那么对任意采样器 $h : \mathcal{I}_u^- \rightarrow l$ ，

$$h^* = \arg \min_{l \in \mathcal{I}_u^-} R(l|i) \quad (4.19)$$

是最小化经验采样风险的最优采样策略。

证明：给定训练集，正样本的分布 $P(i)$ 是确定的。那么经验采样风险是

$$R(h) = \mathbb{E}_{i \sim P(i)} R(h|i) \quad (4.20)$$

其中 $R(h|i)$ 是给定正例 i 的条件采样风险。于是

$$\begin{aligned} R(h^*) - R(h) &= \mathbb{E}_{i \sim P(i)} R(h^*|i) - \mathbb{E}_{i \sim P(i)} R(h|i) \\ &= \sum_i P(i) [R(h^*|i) - R(h|i)] \\ &= \sum_i P(i) [\arg \min_{l \in \mathcal{I}_u^-} R(l|i) - R(h|i)] \\ &\leq 0. \end{aligned} \quad (4.21)$$

因此，经验采样风险的下确界可以表示为最优采样器 h^* 的形式：

$$\begin{aligned} \inf\{R(h)\} &= R(h^*) \\ &= \mathbb{E}_{i \sim P(i)} R(h^*|i). \end{aligned} \quad (4.22)$$

□

证毕。

这个结论是直观的：如果采样器 h^* 最小化了条件采样风险 $R(l|i)$ ，那么经验采样风险也将被最小化。

接下来将估计采样负例实例 l 的采样损失 $\Delta \mathcal{L}(l|i)$ 。为了简化分析，我们遵循^[1]的独立性假设，只考虑单个成对比较损失 $\tilde{\mathcal{L}} = \ln \sigma(\hat{x}_{fn} - \hat{x}_{tn})$ 的变化值。 $\tilde{\mathcal{L}}$ 关于点 \hat{x}_{ul} 的泰勒展开式为：

$$\tilde{\mathcal{L}}' = \begin{cases} \tilde{\mathcal{L}} + \frac{\partial \mathcal{L}}{\partial \hat{x}_{ul}} (\hat{x}'_{ul} - \hat{x}_{ul}) + o(\hat{x}'_{ul} - \hat{x}_{ul}), & \text{if } l \in fn \\ \tilde{\mathcal{L}} - \frac{\partial \mathcal{L}}{\partial \hat{x}_{ul}} (\hat{x}'_{ul} - \hat{x}_{ul}) + o(\hat{x}'_{ul} - \hat{x}_{ul}), & \text{if } l \in tn. \end{cases} \quad (4.23)$$

其中 $\frac{\partial \mathcal{L}}{\partial \hat{x}_{ul}} = \text{info}(l)$ 。因此单位减量 \hat{x}_{ul} (i.e., $\Delta \hat{x}_{ul} = -1$) 导致 $\Delta \mathcal{L}_{fn}(l|i) = \tilde{\mathcal{L}} - \tilde{\mathcal{L}}' \approx \text{info}(l)$, 含义为 $\tilde{\mathcal{L}}$ 减少, 若 l 是伪负例; 否则 $\mathcal{L}_{tn}(l|i) \approx -\text{info}(l)$, 含义为 $\tilde{\mathcal{L}}$ 的增加。为了把所有有序对的 \mathcal{L} 值纳入考虑, 我们引入一个超参数 λ 来控制效果的相对大小, 并估计采样损失为:

$$\Delta \mathcal{L}(l|i) \approx \begin{cases} \text{info}(l), & \text{if } l \in fn \\ -\lambda \text{info}(l), & \text{if } l \in tn \end{cases} \quad (4.24)$$

那么, 给定正例 i , l 的条件采样风险为:

$$R(l|i) = P(fn|l) \cdot \text{info}(l) - P(tn|l) \cdot \lambda \text{info}(l). \quad (4.25)$$

因此, 负采样通过选择条件采样风险最小的样本实现:

$$\begin{aligned} j &= \arg \min_{l \in \mathcal{M}_u} R(l|i) \\ &= \arg \min_{l \in \mathcal{M}_u} [1 - \text{unbias}(l)] \cdot \text{info}(l) - \lambda \cdot \text{unbias}(l) \cdot \text{info}(l) \\ &= \arg \min_{l \in \mathcal{M}_u} \text{info}(l) \cdot [1 - (1 + \lambda) \text{unbias}(l)] \end{aligned} \quad (4.26)$$

其中 $\mathcal{M}_u \subseteq \mathcal{I}_u^-$ 是一个从未交互物品集合 \mathcal{I}_u^- 中随机采样的小的候选集。当 $|\mathcal{M}_u| = |\mathcal{I}_u^-|$ 时, 此时采样器即为经验采样风险最小的理论最优采样器; 当 $\lambda \rightarrow \infty$ 时, h 退化为 $\arg \max_{l \in \mathcal{M}_u} \text{info}(l) \cdot \text{unbias}(l)$, 即采样信息量大 (即困难样本) 且无偏的 (即用户不喜欢) 的样本。由于采样准则的实现依赖于结合了先验信息和样本信息的负信号测度 $\text{unbias}(l)$, 因此称本章提出的方法为贝叶斯负采样算法 (Bayesian Negative Sampling, BNS)。

4.4 算法实现与时间复杂度分析

4.4.1 算法步骤

BNS 也隶属于动态负采样算法的一种, 需要根据样本的预测评分进行采样, 因此抽样分布会随着模型训练而变化, 因此和典型的动态负采样算法一样, BNS 首先

需要计算用户的评分向量。本章提出的负例采样算法实现步骤总结如下：对于候选集合 \mathcal{M}_u 中的每个负例实例：(1) 计算先验概率，(2) 计算经验分布函数 $F(\hat{x}_l)$ 。它反映了模型对未标注样本 l 所属类别的判别。未标注样本 l 的预测评分越高（困难负样本），表明模型认为用户喜欢这个物品， $F(\hat{x}_l)$ 值越接近于 1，即伪负例的概率越高。(3) 计算样本是真负例的后验概率 $\text{unbias}(l)$ 。(4) 根据公式 (4.26) 进行负例采样。它实际上是通过选择一个未标注样本，最大化了下一个训练轮次的 AUC 和当前训练轮次的 AUC 差值。算法 2 给出了提出的负采样算法的伪代码。

4.4.2 时间复杂度

第一步计算先验概率，由于先验概率是静态的，时间复杂度为 $\mathcal{O}(1)$ 。在本章中，使用流行度建模，只需要常数次操作即可实现。第二部计算经验分布函数，时间复杂度为 $\mathcal{O}(|\mathcal{I}|)$ 。第三步计算后验概率，时间复杂度为 $\mathcal{O}(1)$ ；第四步，执行贝叶斯负采样，对于规模为 m 的候选集，需要执行 m 次操作，时间复杂度为 $\mathcal{O}(1)$ 。因此，提出的贝叶斯负采样算法 BNS 相对于物品个数 $|\mathcal{I}|$ 具有线性时间复杂度。

相比于静态负采样算法，以 BNS 为代表的动态负采样算法有额外的计算开销，主要体现在两点：(1) 计算用户的评分向量。从模型训练的角度而言，正向传播实际上只需要计算正负两个样本的预测得分，就可以反向传播更新参数。但是动态负采样由于需要采样困难负样本，所以需要计算其它额外的候选负例评分，从而产生了额外的计算开销。(2) 计算经验分布函数。 m 个候选样本需要计算 m 次经验分布函数。但是需要强调的是，计算用户的评分向量时间复杂度通常远大于计算经验分布函数，前者是矩阵运算，后者是标量运算。

4.5 试验评估

4.5.1 试验设置

4.5.1.1 数据集

本章在三个公共数据集进行了实验，包括 MovieLens-100k, MovieLens-1M 和 Yahoo!-R3^[16]。这些数据集包含用户根据一个离散的五分制评分系统对物品进行评级。跟随^[1]的数据预处理方式，把所有评分转换为隐式反馈。对于每个数据集，随机选择 20% 作为测试数据，其余 80% 作为训练数据。表 4.1 总结了数据集统计信息。

Algorithm 2 贝叶斯负采样算法 (BNS) 伪代码

Input: 交互集合 $\mathcal{R} = \{(u, i)\}$, 评分函数 $s(\cdot)$, 候选集 \mathcal{M}_u 的大小 m , 权重 λ 。

Output: 模型参数 $\Theta \in \mathbb{R}^d$

for $epoch = 1, 2, \dots, T$ **do**

随机采样一个 mini-batch $\mathcal{R}_{batch} \in \mathcal{R}$

for 每个交互 $(u, i) \in \mathcal{R}_{batch}$ **do**

计算评分向量 $\hat{\mathbf{x}}_u$.

\\ 开始负采样

均匀采样候选集 $\mathcal{M}_u \subseteq \mathcal{I}_u^-$.

for 每个候选负例 $(u, l) \in \mathcal{M}_u$ **do**

通过公式(4.4)计算 $info(l)$;

通过公式(4.16)计算 $P_{fn}(l)$; \\ 先验

通过公式(4.15)计算 $F(\hat{x}_l)$; \\ 似然

通过公式(4.14)计算 $unbias(l)$; \\ 后验

根据采样准则(4.26)采样 j ;

更新用户、物品表示 $\mathbf{w}_u, \mathbf{h}_i, \mathbf{h}_j$.

Result: 用户和物品特征表示 Θ 。

表 4.1 数据集统计信息

	用户数	物品数	训练集中交互数	测试集中交互数
MovieLens-100k	943	1,682	80k	20k
MovieLens-1M	6,040	3,952	800k	200k
Yahoo!-R3	5,400	1,000	146k	36k

4.5.1.2 对比算法

对比算法涵盖三种类型的负采样方法: (i) 固定负采样分布, 包括 RNS^[1,5,6,13,14,16] 和 PNS^[52,53,118], (ii) 具有动态采样分布下的硬负例采样 (hard negative sampling), 包括 AOBPR^[45] 和 DNS^[46], 以及 (iii) 基于先验统计信息的采样, 用于过采样高方差负样本, 包括 SRNS^[3]。所有这些对比方法也仅使用 positive-unlabeled 隐式反馈数据, 没有额外的辅助信息用于指导负采样。

- RNS^[1,5,6,13,14,16]: (Random Negative Sampling) 随机均匀采样负例。
- PNS^[52,53,118]: (Popularity-biased Negative Sampling) 依流行度的负采样, 其采样分布正比于物品的交互频率, 即, $\propto r_j^{.75}$ 。
- AOBPR^[45]: 使用采样概率与 $exp(-rank(j|u)/\lambda)$ 成比例的方式对全局排名较高的负样本进行过采样, 其中 $rank(j|u)$ 表示预测得分 \hat{x}_{uj} 在用户 u 的预测得分向量 $\hat{\mathbf{x}}_u$ 中的排名位置, λ 是一个参数。

- DNS^[46]: (Dynamic Negative Sampling) 对于相对排名较高的困难负样本进行过采样。其采样概率是相对排名位置的线性函数。
- SRNS^[3]: 该方法通过统计样本的预测得分，发现伪负例的预测得分具有相对较低的方差。该方法的核心是对高方差负样本进行过采样。

本章使用的推荐模型包，经典的矩阵分解（Matrix Factorization，MF），以及轻量级图卷积神经网络（LightGCN）^[5]。为了公平比较，所有对比的负采样算法设置了相同的推荐模型参数。代码分别使用 Numpy 和 PyTorch 实现。计算是在一台配备 Windows 10 操作系统、2.1 GHz CPU、RTX 1080Ti GPU 和 32 GB RAM 的个人电脑上进行的。实现细节：(a) MF^[121]: 嵌入维度 $d = 32$ ，学习率 $\alpha = 0.01$ ，正则化常数 $reg = 0.01$ ，训练时期 $T = 100$ ，批量大小 $b = 1$ 。(b) LightGCN^[5]: 嵌入维度 $d = 32$ ，初始学习率 $\alpha = 0.01$ ，每 20 个时期衰减一次，衰减率为 0.1，正则化常数 $reg = 10^{-5}$ ，LightGCN 层数 $l = 1$ ，训练轮数 $T = 100$ ，批量大小 $bs = 128$ （对于 MovieLens-100K 和 Yahoo!-R3 数据集）， $bs = 1024$ （对于 MovieLens-1M 数据集）。

4.5.1.3 评估指标

为了评估采样质量，从两个方面衡量采样实例的质量：采样偏倚率（sampling bias rate）和采样信息量，它度量了损失梯度量的大小（average loss gradient magnitude）。通过翻转测试集中真实记录的标签，可以获得在负采样过程中被错误地标记为负样本的伪负例（FN），而未与之交互的其他物品则被视为真负例（TN）。对于每个训练时点，记录每个采样实例的标签和损失梯度幅度 $\text{info}(j)$ ，然后通过如下方式定义时期内的无偏性和信息性：

$$TNR = \frac{\#TN}{\#TN + \#FN}, \quad (4.27)$$

$$INF = \frac{\sum_j \text{info}(j) \cdot \mathbb{I}(j)}{\#TN + \#FN}, \quad (4.28)$$

其中， $\#TN$ ($\#FN$) 是每个训练轮次中采样的真负例（伪负例）的数量。公式 (4.27) 评估了每个训练时期中采样的真负例的比例，即真阴性率（True Negative Rate，TNR）。 $\mathbb{I}(j)$ 是指示函数：如果采样物品的标签是真阴性，则 $\mathbb{I}(j) = 1$ ；否则， $\mathbb{I}(j) = -1$ 作为采样伪负例的惩罚。公式 (4.28) 定义的信息量（Informativeness，INF）可以解释为在每个训练时期中对选定的训练三元组 (u, i, j) 关于梯度的平均幅度。为了评估推荐性能，我们采用了广泛使用的指标，包括准确率（Precision）、召回率（Recall）和

华 中 科 技 大 学 博 士 学 位 论 文

NDCG (归一化折损累积增益), 用于评估 Top- K 推荐。由于这些指标的常见用法, 不在此提供它们的定义。

4.5.2 试验结果

4.5.2.1 推荐性能

表 4.2 Top-k 推荐性能比较

Dataset	CF Model	Method	Top-5			Top-10			Top-20		
			Precision	Recall	NDCG	Precision	Recall	NDCG	Precision	Recall	NDCG
100K	MF	RNS	0.3900	0.1301	0.4143	0.3363	0.2164	0.3967	0.2724	0.3298	0.3962
		PNS	0.2647	0.0864	0.2694	0.2329	0.1475	0.2637	0.1949	0.2374	0.2709
		AOBPR	0.3970	0.1375	0.4186	0.3308	0.2165	0.3942	0.2700	0.3369	0.3980
		DNS	<u>0.4053</u>	<u>0.1414</u>	<u>0.4314</u>	0.3348	<u>0.2214</u>	<u>0.4042</u>	0.2734	<u>0.3413</u>	<u>0.4069</u>
		SRNS	0.3951	0.1342	0.4176	<u>0.3394</u>	0.2174	0.3998	<u>0.2747</u>	0.3374	0.4013
		Proposed	0.4205	0.1467	0.4558	0.3463	0.2290	0.4217	0.2762	0.3466	0.4176
1M	MF	RNS	0.3944	0.1231	0.4204	0.3346	0.2189	0.4017	0.2658	0.3281	0.3986
		PNS	0.3527	0.1266	0.3816	0.3015	0.2117	0.3660	0.2461	0.3306	0.3742
		AOBPR	0.3911	0.1407	0.4200	0.3315	0.2276	0.4007	0.2680	0.3505	0.4064
		DNS	<u>0.4278</u>	<u>0.1475</u>	<u>0.4590</u>	<u>0.3612</u>	<u>0.2336</u>	<u>0.4331</u>	0.2917	<u>0.3595</u>	<u>0.4335</u>
		SRNS	0.4195	0.1440	0.4509	0.3564	0.2333	0.4275	0.2834	0.3520	0.4244
		Proposed	0.4318	0.1518	0.4640	0.3671	0.2410	0.4368	<u>0.2875</u>	0.3608	0.4383
Yahoo	MF	RNS	0.3843	0.0855	0.4027	0.3353	0.1430	0.3737	0.2798	0.2244	0.3572
		PNS	0.3461	0.0753	0.3634	0.3004	0.1250	0.3356	0.2502	0.1979	0.3192
		AOBPR	0.3946	0.0954	0.4135	0.3416	0.1549	0.3837	0.2857	0.2442	0.3714
		DNS	<u>0.4066</u>	<u>0.0991</u>	<u>0.4272</u>	<u>0.3521</u>	<u>0.1620</u>	<u>0.3965</u>	<u>0.2945</u>	<u>0.2537</u>	<u>0.3838</u>
		SRNS	0.3955	0.0934	0.4225	0.3408	0.1609	0.4042	0.2779	0.2431	0.3974
		Proposed	0.4207	0.1062	0.4324	0.3518	0.1703	0.4191	0.3045	0.2614	0.4002
1M	LightGCN	RNS	0.4095	0.0953	0.4305	0.3512	0.1547	0.3985	0.2915	0.2405	0.3781
		PNS	0.3658	0.0907	0.3855	0.3152	0.1486	0.3564	0.2608	0.2314	0.3440
		AOBPR	0.4073	<u>0.0997</u>	0.4286	0.3535	0.1626	0.3982	0.2949	<u>0.2536</u>	<u>0.3849</u>
		DNS	<u>0.4130</u>	0.0972	<u>0.4342</u>	<u>0.3552</u>	0.1577	<u>0.4002</u>	<u>0.2958</u>	0.2468	0.3840
		SRNS	0.4026	0.0973	0.4239	0.3515	0.1526	0.3953	0.2922	0.2524	0.3815
		Proposed	0.4228	0.1087	0.4438	0.3639	<u>0.1612</u>	0.4088	0.3025	0.2527	0.3917
100K	MF	RNS	0.1196	0.0875	0.1326	0.0935	0.1367	0.1401	0.0695	0.2015	0.1665
		PNS	0.1186	0.0876	0.1301	0.0927	0.1360	0.1378	0.0688	0.2011	0.1644
		AOBPR	0.1012	0.0741	0.1115	0.0798	0.1165	0.1184	0.0607	0.1778	0.1443
		DNS	<u>0.1251</u>	<u>0.0917</u>	<u>0.1390</u>	<u>0.0957</u>	<u>0.1399</u>	<u>0.1449</u>	<u>0.0697</u>	0.2020	<u>0.1697</u>
		SRNS	0.1141	0.0855	0.1285	0.0904	0.1358	0.1383	0.0678	<u>0.2025</u>	0.1655
		Proposed	0.1303	0.0975	0.1470	0.1002	0.1485	0.1542	0.0711	0.2094	0.1783
Yahoo	LightGCN	RNS	0.1479	0.1101	0.1693	0.1126	0.1669	0.1760	0.0814	0.2389	0.2047
		PNS	0.1076	0.0797	0.1214	0.0809	0.1185	0.1254	0.0590	0.1708	0.1464
		AOBPR	0.1462	0.1120	0.1635	0.1048	0.1552	0.1612	0.0763	0.2229	0.1886
		DNS	<u>0.1530</u>	<u>0.1137</u>	<u>0.1743</u>	<u>0.1148</u>	<u>0.1697</u>	<u>0.1800</u>	<u>0.0829</u>	<u>0.2433</u>	<u>0.2089</u>
		SRNS	0.1457	0.1092	0.1668	0.1121	0.1636	0.1735	0.0799	0.2352	0.2017
		Proposed	0.1550	0.1157	0.1768	0.1169	0.1729	0.1827	0.0837	0.2459	0.2117

表 4.2 对比了不同负采样算法的 top-k 推荐性能表现, 其中粗体和下划线分别表

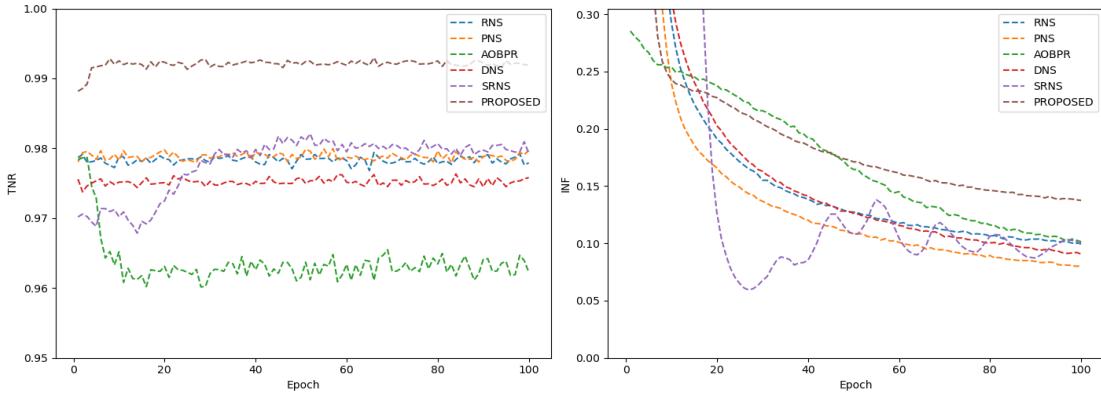


图 4.4 不同方法的采样质量比较

示每个对比组中的最佳和次佳结果。提出的 BNS 算法在两个推荐模型、三个测试数据集和三个性能指标的几乎所有情况下都取得了最佳性能（除了两个次优结果）。值得注意的是，LightGCN 推荐模型普遍优于 MF 模型，这应归功于它使用了图结构和强大的神经模型进行表示学习。

在两种静态负采样算法中，RNS 通常优于 PNS。这表明，基于流行度的负采样实际上可能在负采样中引入更多的偏差，即流行的物品可能会被用户喜欢。在三种困难负采样算法（AOBPR、DNS 和 SRNS）中，DNS 通常优于其他两种算法。AOBPR 优先选择全局排名较高的物品，而 DNS 首先随机选择一些负样本，然后选择于局部排名最高的物品。DNS 这种采样局部排名靠前的算法，有助于在信息性和无偏性之间在一定程度上取得平衡，它在许多情况下能够排名第二。SRNS 利用经验观察，认为真负样本具有较大的分数方差。尽管这是一种有趣的方法来识别真负样本，但 SRNS 选择高质量实例的最终操作是通过信息量和真负例的线性平均，这可能会削弱其负采样算法的有效性。

4.5.2.2 负采样质量

在本小结中，测试了两个采样准则：(i) 由公式(4.14)给出的后验概率准则。(ii) 由公式(4.26)给出的贝叶斯采样准则。前者旨在选择真负样本，而后者旨在采样高质量的负样本实例。我们首先检查后验概率准则是否能够选择真负样本实例，然后检查贝叶斯采样准则是否能够选择高质量的负样本。

后验概率准则通过从候选集 \mathcal{M}_u 中选择具有最大 $\text{unbias}(\cdot)$ 值的负样本实例来

实现：

$$j = \arg \max_{l \in \mathcal{M}_u} \text{unbias}(l) \quad (4.29)$$

\mathcal{I}_u^- 是一个从 \mathcal{I}_u^- 中随机选择的负样本的小候选集，规模大小固定为 5。贝叶斯采样准则通过从候选集 \mathcal{M}_u 中选择具有最小条件采样风险 $R(\cdot|i)$ 的负样本实例来实现：

$$j = \arg \min_{l \in \mathcal{M}_u} \text{info}(l) \cdot [1 - (1 + \lambda)\text{unbias}(l)] \quad (4.30)$$

不同采样方法的采样质量展示在图 4.4 中。

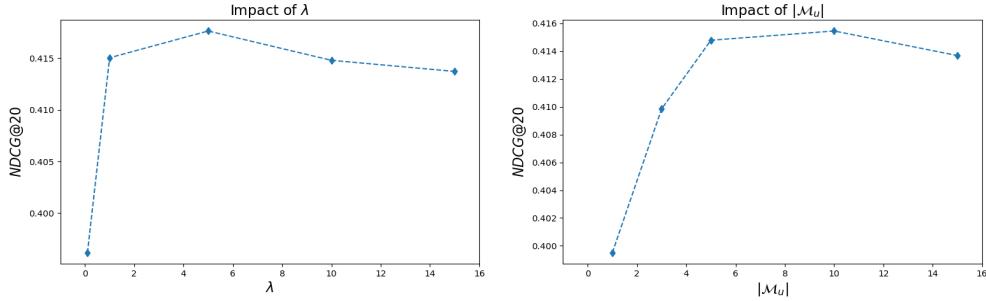
(i) 采样偏差：固定分布采样 (RNS 和 PNS) 表现相对中等。它们的真负样本率 (TNR) 在随机样本为真负样本的概率附近波动。困难负采样 (AOBPR 和 DNS) 表现最差。它们采用贪婪策略强调排名较高的负样本，也带来了更高的采样误差风险，正如我们在第 4.3.1 节中讨论的那样。SRNS 利用预测分数方差的简单先验统计信息，限制了负样本分类的潜力，因为这种先验方差可能导致采样分布过于集中。提出的贝叶斯负采样 (BNS) 由于我们的贝叶斯负分类，TNR 接近 1，实现了最佳性能。

(ii) 采样质量：样本的信息量随训练时期的增加而降低。这是因为经过训练的推荐模型可能会将用户潜在感兴趣的伪负真负样本（参见图 4.2）。在足够的训练时期后，BNS 实现了最佳性能。困难负采样 (AOBPR 和 DNS) 受到最高采样偏差的惩罚更多。SRNS 采用线性加权平均来结合信息性和方差，这可能不能保证采样无偏和信息丰富的实例

4.5.2.3 超参数分析

首先将候选集 \mathcal{M}_u 的大小固定为 5，以研究 λ 对性能的影响。较大的 λ 值意味着更加强调来自真负样本的排名增益，而较小的 λ 值则更加关注避免采样假负样本的风险。从图 4.5 中可以观察到，当 λ 的值从 0.1 增加到 1 时， $NDCG@20$ 显著提高，并且在 $\lambda = 5$ 时达到最大值。

然后，将最优 λ 值固定为 5，研究 \mathcal{M}_u 的大小对性能的影响，并在范围 $\{1, 3, 5, 10, 15\}$ 内搜索 $|\mathcal{M}_u|$ 的取值。需要注意的是，当 $|\mathcal{M}_u| = 1$ 时，提出的 BNS 变为经典的随机负采样 (RNS)。当 $|\mathcal{M}_u| > 1$ 时，贝叶斯采样准则开始发挥其样本筛选的作用。当 $|\mathcal{M}_u| = |\mathcal{I}_u^-|$ 时，公式 (4.26) 中的采样器 h 是由公式 (4.19) 给出的最优采样器 h^* 。从图 4.5 中可以观察到，当 $|\mathcal{M}_u| = 5$ 或 10 时， $NDCG@20$ 达到最大


 图 4.5 参数 λ 和 $|\mathcal{M}_u|$ 的影响

值。为了降低时间复杂度，我们将 $|\mathcal{M}_u| = 5$ 。

当 $|\mathcal{M}_u| > 10$ 时， $NDCG@20$ 的值下降。这个实验结果与预期不符，因为 $|\mathcal{M}_u|$ 值越大，越容易选择最优负样本。我们认为这是由于不可靠的先验概率 $P_{fn}(l)$ 估计引起的。不可靠的先验信息进一步导致了由 $F(\hat{x}_l)$ 表示的分类结果的偏差，进而导致了负信号 $unbias(\cdot)$ 的进一步偏离。过大的 $|\mathcal{M}_u|$ 放大了负信号偏差的不利影响，导致性能下降。更多讨论可以在第 4.5.2.5 节中找到。

本节进一步研究了 BNS 对 λ 、样本信息 \hat{x}_l 和先验信息 $P_{fn}(l)$ 的敏感性。

- **BNS-1:** λ 的热启动。设置 $\lambda = \max(10 - \alpha \times \text{epoch}, 2)$ ，随着 epoch 数的增加，线性降低 λ 的值。即，在初始阶段，较大的 λ 强调对真负样本进行排名增益，而在后期阶段，较小的 λ 强调对假负样本进行采样损失。其中， α 选为 0.1。
- **BNS-2:** 样本信息 \hat{x}_l 的热启动。我们首先采用 RNS 训练一个推荐模型一些 epoch，以学习更可靠的样本信息 $F(\hat{x}_l)$ ，然后通过将其替换为我们的 BNS 来恢复训练。
- **BNS-3:** 无信息先验分布。对于这种情况，BNS 简化为仅使用样本信息 \hat{x}_l 进行负采样的 DNS。对于单次随机试验，任何物品 l 被交互的概率为 $1/1682$ ，即我们对任何负样本 l 无差别地设置 $P_{fn}(l) = 1/1682$ ，其中 1682 是物品的总数。
- **BNS-4:** 职业信息增强的先验分布。在公式 (4.16) 的基础上，我们添加了一个调整因子来改善 $P_{fn}(l)$ 的估计：

$$P_{fn}(l) = \frac{\text{pop}_l}{N} \cdot (1 + \Delta o_{ul}),$$

其中 $\Delta o_{ul} = \frac{o_{ul} - \bar{o}_l}{\max o_l}$ 表示 u 的职业中对 l 的偏好次数与平均值之间的偏差。 o_{ul} 是具有与 u 相同职业的群体在物品 l 上的交互次数， \bar{o}_l 是物品 l 被每个职业交互

华 中 科 技 大 学 博 士 学 位 论 文

的平均次数。

实验结果见表 4.3。以下是我们的发现：

表 4.3 BNS 负采样算法的消融实验

Dataset	CF Model	Method	Top-5			Top-10			Top-20		
			Precision	Recall	NDCG	Precision	Recall	NDCG	Precision	Recall	NDCG
100K	MF	RNS	0.3900	0.1301	0.4143	0.3363	0.2164	0.3967	0.2724	0.3298	0.3962
		BNS	0.4205	0.1467	0.4558	0.3463	0.2290	0.4217	0.2762	0.3466	0.4176
		BNS-1	0.4237	0.1471	0.4551	0.3495	0.2305	0.4238	0.2762	0.3495	0.4197
		BNS-2	0.4148	0.1456	0.4449	0.3411	0.2245	0.4132	0.2738	0.3434	0.4125
		BNS-3	0.4048	0.1392	0.4266	0.3423	0.2282	0.4043	0.2720	0.3406	0.4030
		BNS-4	0.4262	0.1478	0.4566	0.3486	0.2305	0.4235	0.2792	0.3520	0.4216

BNS 对 λ 的敏感性: λ 的热启动取得了更好的性能 (BNS-1)。较大的 λ 值意味着在采样风险和增益之间的权衡程度更高，即更加强调来自真负样本的排名增益而非假负样本的风险。结果显示，对于模型学习来说，困难样本是重要的，这与现有研究的发现一致^[3,59]。建议采用 λ 的热启动策略：在初始阶段使用较大的值强化对困难样本的采样，而在后期阶段使用较小的 λ 以避免采样伪负例。

BNS 对先验概率 $P_{fn}(\cdot)$ 的敏感性: 结果显示，在没有先验信息的情况下，BNS-3 相对于标准的 BNS 性能较差，而在增强职业信息先验概率的情况下，BNS-4 相对于标准的 BNS 性能更好。结果表明，BNS 对先验概率敏感。先验信息影响负采样的机制是：不可靠的先验概率 $P_{fn}(\cdot)$ 导致由 $F(\hat{x})$ 表示的分类结果偏差，进而导致负信号 $unbias(\cdot)$ 的进一步偏离。过大的候选集 \mathcal{M}_u 放大了负信号偏差的不利影响，导致性能下降。因此，如果先验概率 $P_{fn}(\cdot)$ 可靠，则选择更大的 \mathcal{M}_u 更好；否则，应选择适度大小的 \mathcal{M}_u 。特别地，在非信息先验分布的情况下，BNS 等效于 DNS：BNS 采样具有适当 $F(\hat{x})$ 值的实例，而 DNS 采样具有适当排名位置的实例，而 $F(\hat{x})$ 和排名位置具有一对一的映射关系。通过选择适当的 $|\mathcal{M}_u|$ 和 λ ，BNS-3 的性能与 DNS 相当。我们建议选择最可靠的先验信息来建模 $P_{tn}(\cdot)$ 或 $P_{fn}(\cdot)$ 。

BNS 对样本信息 \hat{x}_l 的敏感性: 样本信息 \hat{x} 的热启动 (BNS-2) 并没有像预期的那样取得更好的性能，我们认为有三个原因：(i) 初始训练阶段的随机采样难以采样到难负例，导致 BNS-2 的性能下降；(ii) \hat{x} 是由先验概率和采样器 h 内生确定的，因此 \hat{x} 的热启动对最终的排名性能影响有限；(iii) 我们使用样本信息 $F(\hat{x})$ 的方式对 \hat{x} 的微小变化不敏感，因此改进的 \hat{x} 对提高采样质量的影响有限。我们认为 BNS 具有

对不同排名模型的鲁棒性，这是它对不同排名模型具有鲁棒性的重要体现，因为在不满足顺序关系的条件下（例如早期训练阶段或弱排名模型），BNS 仍然能够利用先验信息采样高质量的负实例。

4.5.2.4 向其他损失函数推广

本小结检验 BNS 在其他损失函数中的适用性。我们选择了两个基于对比的损失函数，即二元交叉熵（BCE）损失和 InfoNCE 损失函数。我们使用矩阵分解模型作为编码器，并分别将 BPR 损失^[1] 替换为 BCE 损失^[122] 和 Info-NCE^[7] 损失。超参数保持与使用 BPR 损失时相同。具体而言，InfoNCE 损失的负实例数量 N 固定为 2。表 4.4 展示了使用不同损失函数的结果。可以看出，与随机负采样 RNS 相比，BNS 在所有三个损失函数中都取得了显著的改进，展示了 BNS 的良好适用性。

表 4.4 不同损失函数下 BNS 负采样算法的表现

Dataset	CF Model	Loss Function Method	Top-5			Top-10			Top-20			
			Precision	Recall	NDCG	Precision	Recall	NDCG	Precision	Recall	NDCG	
100K	MF	BPR	RNS	0.3900	0.1301	0.4143	0.3363	0.2164	0.3967	0.2724	0.3298	0.3962
			BNS	0.4205	0.1467	0.4558	0.3463	0.2290	0.4217	0.2762	0.3466	0.4176
		BCE	RNS	0.3917	0.1329	0.4107	0.3339	0.2187	0.3923	0.2696	0.3315	0.3928
			BNS	0.4226	0.1454	0.4490	0.3514	0.2299	0.4205	0.2821	0.3482	0.4182
		Info NCE	RNS	0.3923	0.1322	0.4193	0.3353	0.2162	0.3988	0.2728	0.3376	0.4012
			BNS	0.4241	0.1471	0.4523	0.3519	0.2281	0.4226	0.2817	0.3467	0.4195

4.5.2.5 渐近最优采样器

本小结展示所提出的采样器 h 通过理想先验概率 $P_{fn}(l)$ 以渐进最优采样器 h^* 的过程。设置 $P_{fn}(l) = (\text{label}(l) - 0.2)^2$ ，即如果 $l \in fn$ ，则 $P_{fn}(l) = 0.64$ ，否则 $P_{fn}(l) = 0.04$ 。通过逐渐增加 \mathcal{M}_u 的大小，以渐近最优采样器，其中 λ 固定为 5。模拟结果如表 4.5 所示。通过增加候选集的大小，可以实现最优采样器 h^* 而不会降低排名性能。该结果验证了我们在第 4.5.2.3 节中的分析。在具备一定程度的先验信息的情况下，即使对于简单的基于点乘的表示学习方法，BNS 也能取得可观的性能，展示了负采样研究的巨大潜力。最优采样器（即 $|\mathcal{M}_u| = |\mathcal{I}_u^-|$ ）的性能是基于点乘模型的经验上界。由于存在低秩约束和矩阵分解的有限表达能力，推荐性能无法达到 1。

这些基准结果包括两个有启发性的消融实验的结果：(i) 只使用先验信息（物品流行度）的 PNS，以及 (ii) 只使用样本信息 $\hat{x}l$ 的 DNS、AOBPR 和 SRNS。特别地，在无信息的先验分布的情况下，BNS 退化为 DNS。BNS 根据适当的 $F(\hat{x})$ -值对实例

表 4.5 向最优采样器的渐进过程

Dataset	CF Model	BNS Size	Top-5			Top-10			Top-20		
			Precision	Recall	NDCG	Precision	Recall	NDCG	Precision	Recall	NDCG
100K	MF	$ \mathcal{M}_u = 1$	0.3900	0.1301	0.4143	0.3363	0.2164	0.3967	0.2724	0.3298	0.3962
		$ \mathcal{M}_u = 3$	0.4909	0.1567	0.5211	0.4220	0.2565	0.4942	0.3366	0.3872	0.4856
		$ \mathcal{M}_u = 5$	0.5109	0.1612	0.5422	0.4329	0.2602	0.5092	0.3456	0.3925	0.4992
		$ \mathcal{M}_u = 10$	0.5351	0.1696	0.5685	0.4589	0.2722	0.5365	0.3663	0.4081	0.5245
		$ \mathcal{M}_u = 20$	0.5760	0.1828	0.6070	0.4885	0.2875	0.5695	0.3830	0.4196	0.5498
		$ \mathcal{M}_u = 50$	0.6239	0.1989	0.6599	0.5252	0.3049	0.6146	0.4031	0.4312	0.5843
		$ \mathcal{M}_u = 100$	0.6509	0.2104	0.6898	0.5382	0.3125	0.6346	0.4053	0.4321	0.5971
		$ \mathcal{M}_u = 500$	0.6661	0.2183	0.7128	0.5412	0.3131	0.6487	0.4041	0.4300	0.6076
		$ \mathcal{M}_u = \mathcal{I}_u^- $	0.6674	0.2184	0.7133	0.5429	0.3140	0.6495	0.4041	0.4292	0.6073

进行采样，而 DNS 根据适当的排名位置对实例进行采样，由于 $F(\hat{x})$ 和排名位置具有一对一的映射关系，因此它们实现了可比较的性能。上述两种范式的缺点是显而易见的：前者可以融合领域知识，但独立于模型状态，导致了静态的采样分布；后者利用了样本信息，但忽略了先验知识，在富有辅助信息的场景中，这将导致性能下降。所提出的 BNS 从贝叶斯的角度结合了先验信息和样本信息，并以赋予负信号 $\text{unbias}(l)$ 以后验概率的内涵。本章认为 BNS 的性能改进源于三个方面：(i) 先验信息 $P_{tn}(l)$ 或 $P_{fn}(l)$ ，以及 (ii) 使用样本信息 \hat{x}_l 。(iii) BNS 是最小化经验采样风险的理论最优采样器。

4.6 本章小结

本章定义了一个与模型无关的后验概率估计，作为定量的负信号度量。而后提出了贝叶斯负采样规则，这是最小化经验采样风险的理论最优的采样规则。BNS 的贡献在于以后验概率的意义上指定了负信号度量。从贝叶斯的角度，BNS 通过结合静态的先验信息和动态的样本信息，统一了现有的两种提取负信号的范式。需要注意的是，BNS 并未具体指定建模先验概率的方法。相反， $P_{tn}(l)$ （或 $P_{fn}(l)$ ）为使用辅助信息建模先验概率的方法提供了接口。此外，只要模型旨在将正例的得分高于负例，BNS 也可以应用于其他基于对比的损失函数。

需要指出的是，负采样算法与基于 GPU 批量并行计算 Mini-Batch 模型并不兼容。典型的动态负采样需要先计算当前轮次样本得分（正向传播），然后采样负例。而基于 Mini-Batch 模型需要把样本固定到 DataLoader 中，才能进行正向传播预测得分，这与动态负采样机制有所冲突。而动态负采样需要先得到预测得分，然后再采

华 中 科 技 大 学 博 士 学 位 论 文

样负例。通常是把很多的候选负样本装载入 Dataloader，并计算额外样本的预测得分来实现动态负采样，造成了额外的计算和存储开销。此外，本章所假设的真负例和伪负例的相似度分数的序关系是一个理想的情况，是一个比较强的假设。后面的章节将解决这两个局限性。

5 成对损失函数校正研究

从成对比较中学习对比表示 (contrastive representations) 在自然语言处理、计算机视觉和信息检索等多个领域取得了显著的成功。基于成对学习的协同过滤算法也根植于这一范式。一个重要的问题是数据集普遍是 positive-unlabeled，即负例未标注，这经常导致随机选择的负例包含了伪负例，并且不可避免地引入导致学到有偏的嵌入，从而影响下游的分类、排序任务的泛化性能。上一章介绍了最优负采样准则，但是负采样算法不适用于基于 GPU 批量并行计算的 mini-batch 模型，往往以额外的存储开销和计算开销为代价实现。本章采用了一个截然不的技术路径纠正采样偏差，得到了一种校正后的成对学习损失函数，称为去偏成对损失 (Debiased Pairwise Loss, DPL)。DPL 的核心思想是纠正由于伪负例引起的偏倚概率估计，从而修正梯度，使其逼近完全监督数据的梯度。对五个公共数据集进行的实验研究验证了所提出的学习方法的有效性。

5.1 引言

成对学习鼓励编码器对样本之间的差异特征进行编码，而不是对个体样本的像素级特征进行编码，通常能够获得更好的泛化性能，尤其是在样本的绝对值较少有意义的场景下 [21,25,26,32,32]。成对学习已成为许多现代机器学习算法的基本组成部分，并在自然语言处理、图像和语音识别以及推荐系统等诸多领域取得了重大进展 [1,7,9,32]。在协同过滤中，成对学习被广泛应用于预测排序，其中最著名的方法是贝叶斯个性化排名 (BPR) [1]。从统计角度来看，BPR 最大化了观察到的正负样本的后验概率。从数值计算角度来看，BPR 损失函数鼓励模型正例的得分高于负例。在嵌入空间中，BPR 损失的目标是将正例的嵌入拉近到锚点（即用户），同时将负例的嵌入推离锚点。

然而，用户通常只通过交互行为（如点击、购买或评分）来表达他们的偏好或兴趣，从而仅提供正反馈。因此，训练集通常以正无标签 (PU) 数据的形式存在。针对这一在机器学习的诸多领域普遍存在的问题，负采样技术被广泛研究。负采样可以分为两种类型：第一种是静态负采样，它使用固定的采样分布，利用与模型的训练

状态无关的某种辅助信息，可能导致简单的样本，相比之下，与动态负采样相比，静态负采样的性能往往较差。此外，静态负采样严重依赖于辅助信息作为有效的监督信号，而这些监督信息在大多数场景下是难以获取的。动态负采样使用与模型相关的信息（如预测得分）动态调整采样分布，旨在采样具有高得分或排名靠前的困难负样本，以提高性能，但容易遇到伪负例^[3]。此外，基于 GPU 批量计算的 mini-batch 训练与动态负采样不兼容。矛盾的关键在于，mini-batch 训练要求训练之前将样本固定加载到数据加载器中，然后进行正向预测得分和反向传播更新参数；而动态负采样要求先正向预测得分，然后采样样本。动态负采样通常需要额外的计算和存储开销，例如存储先前训练轮次的预测得分^[3]，或计算 mini-batch 之外样本的预测得分。

本章聚焦于没有任何辅助信息用于监督的最一般形式的隐式反馈数据，提出了一种从无标签数据中纠正采样偏差的方法，从而为成对学习提供了一种修改后的损失函数，称为去偏成对损失（Debiased Pairwise Loss, DPL）。DPL 的核心思想是修正由于假负例导致的概率估计偏差，从而修正梯度以近似完全监督数据的梯度。DPL 不需要额外的辅助信息进行监督，也不需要过多的存储和计算开销。

5.2 成对损失优化目标分析

将用户物品对 (u, i) 表示为样本 \mathbf{x} ，其中 $u \in \mathcal{U}$, $i \in \mathcal{I}$ 。令 $\mathcal{X} = \{\mathbf{x} | u \in \mathcal{U}, i \in \mathcal{I}\}$ 表示包含所有用户物品对的样本空间， $\mathcal{Y} = \{-1, +1\}$ 表示类别标签，指示用户是否喜欢该物品。决策函数 $g : \mathcal{X} \rightarrow \mathbb{R}$ 是一个实值函数，为每个交互分配一个预测的偏好水平 $g(\mathbf{x}) \in \mathbb{R}$ 。将正例的类条件密度表示为 $p^+(\mathbf{x}) = p(\mathbf{x}|+1)$ ，将负例的类条件密度表示为 $p^-(\mathbf{x}) = p(\mathbf{x}|-1)$ 。因此，边际分布 $p(\mathbf{x}) = p^+(\mathbf{x})\tau^+ + p^-(\mathbf{x})\tau^-$ ，其中 $\tau^+ = 1 - \tau^-$ 是先验概率 $p(c(\mathbf{x}) = +1)$ 。

在隐式协同过滤中，通过对两个随机样本 $(\mathbf{x}^+, \mathbf{x}^-)$ 进行成对比较，通过预测的偏好水平 $g(\mathbf{x}) \in \mathbb{R}$ 以预测个性化排序。BPR^[1] 采用了著名的 Bradley-Terry 模型建模观察到正例 \mathbf{x}^+ 优于负例 \mathbf{x}^- 的似然

$$\begin{aligned} \mathcal{L}_{BPR} &= -\mathbb{E}_{\substack{\mathbf{x}^+ \sim p^+(\mathbf{x}) \\ \mathbf{x}^- \sim p^-(\mathbf{x})}} \log \sigma(g(\mathbf{x}^+) - g(\mathbf{x}^-)) \\ &= -\mathbb{E}_{\substack{\mathbf{x}^+ \sim p^+(\mathbf{x}) \\ \mathbf{x}^- \sim p^-(\mathbf{x})}} \log \frac{1}{1 + \exp(-g(\mathbf{x}^+) + g(\mathbf{x}^-))} \end{aligned} \quad (5.1)$$

$$= -\mathbb{E}_{\substack{\mathbf{x}^+ \sim p^+(\mathbf{x}) \\ \mathbf{x}^- \sim p^-(\mathbf{x}^-)}} \log \frac{\exp(g(\mathbf{x}^+))}{\exp(g(\mathbf{x}^+)) + \exp(g(\mathbf{x}^-))} \quad (5.2)$$

值得注意的是，式(5.2)是贝叶斯个性化排名（BPR）损失的等价形式，它与噪声对比估计（NCE）损失^[24]完全相同，而 NCE 损失是 InfoNCE 损失^[7]在只有一个负例（即 $N=1$ ）的特殊情况。此外，在协同过滤的场景中，用户和物品构成一个二部图，通常只选择用户嵌入作为锚点。在理想情况下，正例是从已有交互的物品中进行采样的，表示为 $\mathbf{x}^+ \in \mathcal{D}^+$ ，而负例是从用户不喜欢的物品中进行采样的，表示为 $\mathbf{x}^- \in \mathcal{D}^-$ 。因此，式(5.1)的对应的经验估计可以表示为：

$$\mathcal{L}_{\text{BPR}} = -\frac{1}{|\mathcal{D}^+| \times |\mathcal{D}^-|} \sum_{\mathbf{x}^+ \in \mathcal{D}^+} \sum_{\mathbf{x}^- \in \mathcal{D}^-} \ln \sigma(g(\mathbf{x}^+) - g(\mathbf{x}^-)) - \lambda \|\Theta\|^2,$$

其中， $\lambda \|\Theta\|^2$ 是正则化项，用于平衡方差和偏差，避免过拟合。特别地，正则化项 $\lambda \|\Theta\|^2$ 等价于高斯分布的先验密度的对数，从而为式(5.3)提供了基于后验概率的解释。在原始的贝叶斯个性化排名（BPR）论文^[1]中，式(5.3)被解释为观测到的有序对的最大后验估计。

在实际中，由于缺乏标记的负例样本，在优化式(5.3)时，只能从未标记的数据中采样负例样本，导致以下偏倚的优化目标：

$$\mathcal{L} = -\frac{1}{|\mathcal{D}^+| \times |\mathcal{D}^u|} \sum_{\mathbf{x}^+ \in \mathcal{D}^+} \sum_{\mathbf{x} \in \mathcal{D}^u} \ln \sigma(g(\mathbf{x}^+) - g(\mathbf{x})) - \lambda \|\Theta\|^2, \quad (5.3)$$

接下来，考察偏倚的优化目标对学习到的用户-物品嵌入 Θ 的影响：

$$\frac{\partial \mathcal{L}}{\partial \Theta} = \frac{\partial \mathcal{L}_{\text{BPR}}}{g(\mathbf{x})} \cdot \frac{g(\mathbf{x})}{\Theta} \quad (5.4)$$

$$= [1 - \sigma(g(\mathbf{x}^+) - g(\mathbf{x}))] \cdot \frac{g(\mathbf{x})}{\Theta} \quad (5.5)$$

式(5.5)是微分链式法则的结果，其中第一项 $[1 - \sigma(g(\mathbf{x}^+) - g(\mathbf{x}))]$ 由损失函数的形式决定，第二项 $\frac{g(\mathbf{x})}{\Theta}$ 由决策函数决定。对于固定的模型，第二项保持不变。

在式(5.5)的第一项中，实值 sigmoid 函数 $\sigma(g(\mathbf{x}^+) - g(\mathbf{x})) \in [0, 1]$ 被解释为用户偏好正例强于负例的似然^[1]，是优化目标。然而，由于 \mathbf{x} 是未标记样本，它以 τ^+ 的先验概率是正样本，这导致了概率的偏倚估计。

图5.1提供了一个阐释性的例子说明为何使用正例-未标记数据对计算的值 $\sigma(g(\mathbf{x}^+) - g(\mathbf{x}))$ 是一个偏倚的概率估计。“偏好正例强于负例的似然”，描述的是事件

$\mathcal{A}_1(+, -)$ 的概率, 这是优化目标。然而在实践中, 给出的是正例-未标注样本对, 导致成对学习优化的是事件 $\mathcal{A}(+, u)$ 的概率。然而, 事件 $\mathcal{A}(+, u)$ 的概率和事件 $\mathcal{A}_1(+, -)$ 的概率是不等价的, 根据事件的可列可加性原理, 事件 $\mathcal{A}(+, u) = \mathcal{A}_1(+, -) + \mathcal{A}_2(+, +)$ 。这个偏差, 导致了偏倚的“偏好正例强于负例”的概率估计, 从而导致了有偏的最大似然/后验的优化目标。直观地说, 偏倚的 $\sigma(g(\mathbf{x}^+) - g(\mathbf{x}))$ 值会导致不正确的梯度大小 $\partial \mathcal{L}_{\text{BPR}} / \partial \Theta$, 从而在执行随机梯度下降学习算法时导致不准确的用户-物品表示。

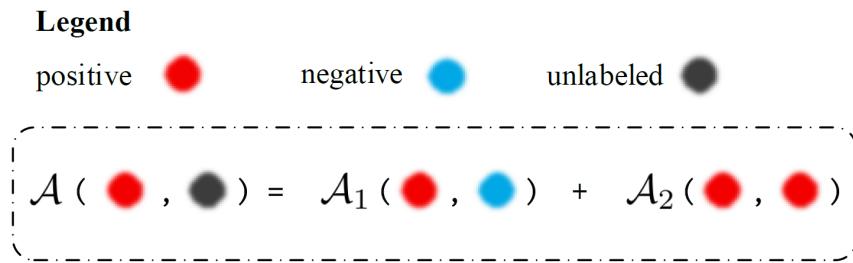


图 5.1 偏倚的概率估计的示意图

为了使数学表达中保持简洁, 本章定义一个映射 $h : \mathcal{X} \times \mathcal{X} \rightarrow \sigma(g(\mathbf{x}^+) - g(\mathbf{x}))$, 将两个样本映射到一个 sigmoid 函数。问题变成如何使用来自正例和未标记样本群体的样本来近似估计 $h(\mathbf{x}^+, \mathbf{x}^-)$ 的值。具体而言, 给定一组正例样本 $\{\mathbf{x}_i\}_{i=1}^M$ 和一组未标记样本 $\{\mathbf{x}_j\}_{j=1}^N$, 目标是估计 $h(\mathbf{x}^+, \mathbf{x}^-)$ 的值, 从而可以校正梯度以近似完全监督数据的梯度, 从而提高学习到的用户-物品表示的泛化性能。

5.3 去偏成对学习算法

为了使用来自正例-未标记样本对来近似估计 $h(\mathbf{x}^+, \mathbf{x}^-)$ 的值, 首先建立正例-未标记样本对的联合分布 p_{PU} 与正例-负例样本对的联合分布 p_{PN} 之间的关系。

样本对的第一个样本, 记为 \mathbf{x}_1 , 是从正类条件概率 $p^+(\mathbf{x})$ 中确定性地抽取的, 而第二个样本, 记为 \mathbf{x}_2 , 是从边缘分布 $p(\mathbf{x})$ 中抽取的。因此, 正例-未标记样本对的联合分布 p_{PU} 可以表示如下:

$$p_{\text{PU}}(\mathbf{x}_1, \mathbf{x}_2) = p^+(\mathbf{x}_1)p(\mathbf{x}_2) \quad (5.6)$$

$$= p^+(\mathbf{x}_1)[p^+(\mathbf{x}_2)\tau^+ + p^-(\mathbf{x}_2)\tau^-] \quad (5.7)$$

$$= \tau^+ p^+(\mathbf{x}_1) p^+(\mathbf{x}_2) + \tau^- p^+(\mathbf{x}_1) p^-(\mathbf{x}_2) \quad (5.8)$$

式(5.6)是由于 $(\mathbf{x}_1, \mathbf{x}_2)$ 是独立抽取的。此外，式(5.7)是边缘分布 $p(\mathbf{x})$ 的完整概率分解。重新整理式(5.8)，可以建立所需的正例-负例样本对的联合分布 $p_{PN}(\mathbf{x}_1, \mathbf{x}_2)$ 与正例-未标记样本对的联合分布 $p_{PU}(\mathbf{x}_1, \mathbf{x}_2)$ 之间的关系，表示为

$$\begin{aligned} p_{PN}(\mathbf{x}_1, \mathbf{x}_2) &= p^+(\mathbf{x}_1) p^-(\mathbf{x}_2) \\ &= \frac{1}{\tau^-} [p_{PU}(\mathbf{x}_1, \mathbf{x}_2) - \tau^+ p^+(\mathbf{x}_1) p^+(\mathbf{x}_2)] \end{aligned} \quad (5.9)$$

因此， $h(\mathbf{x}_1, \mathbf{x}_2)$ 相对正例-负例样本对的联合分布 $p_{PN}(\mathbf{x}_1, \mathbf{x}_2)$ 上的期望，即理想的优化目标“用户喜欢正例 \mathbf{x}_1 优于负例 \mathbf{x}_2 ”的概率期望值，可以计算如下：

$$\mathbb{E}_{p_{PN}(\mathbf{x}_1, \mathbf{x}_2)} h(\mathbf{x}_1, \mathbf{x}_2) \quad (5.10)$$

$$= \int_{\mathbf{x}_1} \int_{\mathbf{x}_2} h(\mathbf{x}_1, \mathbf{x}_2) p_{PN}(\mathbf{x}_1, \mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2 \quad (5.11)$$

$$= \int_{\mathbf{x}_1} \int_{\mathbf{x}_2} h(\mathbf{x}_1, \mathbf{x}_2) [\frac{1}{\tau^-} p_{PU}(\mathbf{x}_1, \mathbf{x}_2) - \frac{\tau^+}{\tau^-} p^+(\mathbf{x}_1) p^+(\mathbf{x}_2)] d\mathbf{x}_1 d\mathbf{x}_2 \quad (5.12)$$

$$= \int_{\mathbf{x}_1} \int_{\mathbf{x}_2} h(\mathbf{x}_1, \mathbf{x}_2) [\frac{1}{\tau^-} p_{PU}(\mathbf{x}_1, \mathbf{x}_2) d\mathbf{x}_1 d\mathbf{x}_2 - \int_{\mathbf{x}_1} \int_{\mathbf{x}_2} \frac{\tau^+}{\tau^-} p^+(\mathbf{x}_1) p^+(\mathbf{x}_2)] d\mathbf{x}_1 d\mathbf{x}_2 \quad (5.13)$$

$$= \frac{1}{\tau^-} \mathbb{E}_{p_{PU}(\mathbf{x}_1, \mathbf{x}_2)} h(\mathbf{x}_1, \mathbf{x}_2) - \frac{\tau^+}{\tau^-} \mathbb{E}_{p_{PP}(\mathbf{x}_1, \mathbf{x}_2)} h(\mathbf{x}_1, \mathbf{x}_2) \quad (5.14)$$

式(5.14)的第一项表示正例-未标记样本对的期望，在给定第一个正例样本 \mathbf{x}^+ 的情况下，可以使用 N 个未标记样本 $\{\mathbf{x}_n\}_{n=1}^N$ 进行经验估计：

$$\mathbb{E}_{p_{PU}(\mathbf{x}_1, \mathbf{x}_2)} h(\mathbf{x}_1, \mathbf{x}_2) = \frac{1}{N} \sum_{n=1}^N h(\mathbf{x}^+, \mathbf{x}_n), \quad (5.15)$$

式(5.14)的第二项表示正例-正例样本对的期望，在给定第一个正例样本 \mathbf{x}^+ 的情况下，可以使用额外的 M 个正例样本 $\{\mathbf{x}'_m\}_{m=1}^M$ 进行经验估计

$$\mathbb{E}_{p_{PP}(\mathbf{x}_1, \mathbf{x}_2)} h(\mathbf{x}_1, \mathbf{x}_2) = \frac{1}{M} \sum_{m=1}^M h(\mathbf{x}^+, \mathbf{x}'_m). \quad (5.16)$$

将式(5.15)和式(5.16)代入式(5.14)，可以得到理想的优化目标“用户喜欢正例 \mathbf{x}_1 优于负例 \mathbf{x}_2 ”的概率期望值的经验估计：

$$\hat{P}_{PN} = \frac{1}{N\tau^-} \sum_{\mathbf{x} \in \mathcal{D}^u} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{M\tau^-} \sum_{\mathbf{x}' \in \mathcal{D}^+} h(\mathbf{x}^+, \mathbf{x}') \quad (5.17)$$

如上所示，方程中的第一项使用正例-未标记样本对进行计算，但额外添加了一个校正项，以抵消由于未标记集合中包含错误的负例样本而产生的偏倚似然估计。式(5.17)中的第二项，从正例-正例对计算的校正项可能看起来奇怪，图5.2对这个校正项提供了一个直观解释。

考虑一个正例-未标记的数据集，其中 $\{x_1, x_2\}$ 是正例样本， $\{x_3, x_4, x_5\}$ 是未标记样本。在训练过程中，无法访问未标记样本 $\{x_3, x_4, x_5\}$ 的真实标签。式(5.17)的第一项描述的是使用正例样本和未标记数据对计算的 AUC 风险，即把不可微的 0-1 损失替换为可微的 sigmoid 损失所得到的 AUC 值，记为 AUC_{PU} 。根据 AUC 的定义， AUC_{PU} 可以分解为两个项的和组成。第一个项是使用正例-负例 (PN) 数据对计算的干净 AUC 估计，表示为 AUC_{PN} 。第二个项是使用正例-正例 (PP) 数据对计算的伪 AUC 估计，表示为 AUC_{PP} 。为了优化干净 AUC，应该从 AUC_{PU} 中减去 AUC_{PP} 。而减去的补偿项目，正是由式(5.17)的第二项基于正例-正例对计算的伪 AUC 估计。

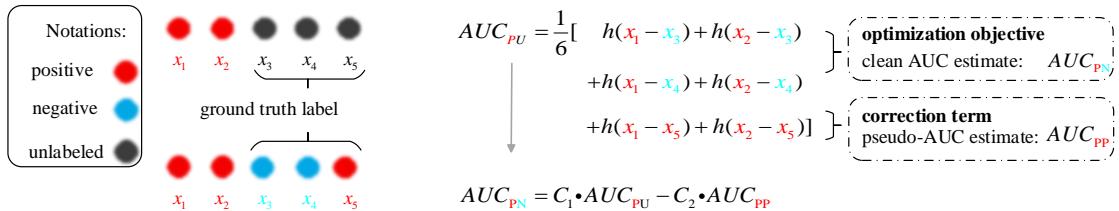


图 5.2 去偏成对损失的示意图

有了式(5.17)给出的优化目标“用户喜欢正例 x_1 强于负例 x_2 ”的估计，可以很容易进行极大似然估计。按照 BPR 的做法，本章也最大化对数似然，那么去偏的成对损失 (Debiased Pairwise Loss, DPL) 的最终经验形式如下所示：

$$\mathcal{L}_{DPL} = -\frac{1}{|\mathcal{D}^+| \times |\mathcal{D}^u|} \sum_{\mathbf{x}^+ \in \mathcal{D}^+} \sum_{\mathbf{x} \in \mathcal{D}^u} \ln \hat{P}_{PN}. \quad (5.18)$$

5.4 算法实现与时间复杂度分析

在使用 BPR 损失进行训练时，对于每个 (u, i) 对，需要一个负样本 j ，从而形成一个 (u, i, j) 的训练三元组。为了解决采样偏差，DPL 损失要求对于每个 (u, i) 对，额外采样 $M \geq 1$ 个正例和 $N \geq 1$ 个负例。遵循与 BPR 相同的数据条

华 中 科 技 大 学 博 士 学 位 论 文

目格式，每个 DPL 数据条目被组织为 $(u, i, i_1, i_2, \dots, i_M, j_1, j_2, \dots, j_N)$ 。这可以通过重写Dataloader的collate_fn函数，实现将一个批量大小为 bs 个交互 (u, i) 的 mini-batch 数据，改写为含有 bs 个数据条目为 $(u, i, i_1, i_2, \dots, i_M, j_1, j_2, \dots, j_N)$ 的 mini-batch 数据。算法伪码如下：

Algorithm 3 Pytorch 的 collate_fn 函数重写

Input: 含有 bs 个交互 (u, i) 的 mini-batch 数据 \mathcal{R} ，批量大小 bs ，物品集合 \mathcal{I} ，用户正例集合 \mathcal{I}_u^+ 。

Output: 含有 bs 个条目为 $(u, i, i_1, i_2, \dots, i_M, j_1, j_2, \dots, j_N)$ 的 mini-batch 数据 \mathcal{R}' 。

for $(u, i) \in \mathcal{R}$ **do**

 从正例集合 \mathcal{I}_u^+ 中随机采样 M 个正例 i_1, i_2, \dots, i_M ;

 从未标注样本集合 \mathcal{I}_u^+ 中随机采样 N 个未标注样本 j_1, j_2, \dots, j_N ;

 将交互 (u, i) 、 M 个正例、 N 个负例拼接为 1 个数据条目；

 将该数据条目添加到 \mathcal{R}' 中；

Result: 含有 bs 个条目为 $(u, i, i_1, i_2, \dots, i_M, j_1, j_2, \dots, j_N)$ 的 mini-batch 数据 \mathcal{R}' 。

因此，每个 mini-batch 数据结构组织如下：

$$\text{batch size} \left\{ \begin{bmatrix} u^1 & i & i_1 & i_2 & \dots & i_M & j_1 & j_2 & \dots & j_N \\ u^2 & i & i_1 & i_2 & \dots & i_M & j_1 & j_2 & \dots & j_N \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ u^{bs} & i & i_1 & i_2 & \dots & i_M & j_1 & j_2 & \dots & j_N \end{bmatrix} \right. \quad (5.19)$$

每个数据条目包括 N 个未标记的样本 j_1, j_2, \dots, j_N ，可以计算相应的 N 个预测得分 $\hat{x}j^1, \hat{x}j^2, \dots, \hat{x}j^N$ 。对于给定的正例样本 (u, i) ，其得分为 \hat{x}_i ，可以基于 N 个负例样本的预测得分得到 N 个 PU 概率值。因此，基于 mini-batch 估计的 PU 概率值为

$$\hat{P}_{\text{pu}} = \frac{1}{N} \sum_{n=1}^N \sigma(\hat{x}_i - \hat{x}_j^n)$$

类似地，可以计算出 M 个正例 i_1, i_2, \dots, i_M 对应的 M 个预测得分 $\hat{x}i^1, \hat{x}i^2, \dots, \hat{x}_i^M$ ，相应地，计算 PP 概率值估计为：

$$\hat{P}_{\text{pp}} = \frac{1}{M} \sum_{m=1}^M \sigma(\hat{x}_i - \hat{x}_i^m)$$

因此，经过校正后的理想的优化目标为：

$$\hat{P}_{\text{PN}} = \frac{1}{\tau^-} \hat{P}_{\text{PU}} - \frac{\tau^+}{\tau^-} \hat{P}_{\text{PP}}$$

和 BPR 一样最大化对数似然，只需要对上式求对数后，执行梯度下降，基于 Pytorch 矩阵运算的算法伪码如下：

Algorithm 4 去偏成对学习算法 (DPL) 伪代码

Input: Mini-batch 数据 \mathcal{R}' 如式(5.19)所示, 评分函数 $s(\cdot)$, 额外正例数量 M , 额外负例数量 N , 正例类先验 τ^+ , 批量大小 bs 。

Output: 模型参数 $\Theta \in \mathbb{R}^d$

for 迭代次数 $t = 1, 2, \dots, T$ **do**

```

从训练集中抽样一个 mini-batch 数据  $\mathcal{R}'$  如式(5.19)所示;
正向计算评分 scores = s( $\mathcal{R}'$ ); # [bs * (1+M+N)]
pos_scores = scores[:, : M+2]; # [bs * (1+M)]
neg_scores = scores[:, M+2:]; # [bs * N]
pu_prob = sigmoid(pos_scores[:, 0:1] - neg_scores); # [bs*N]
pp_prob = sigmoid(pos_scores[:, 0:1] - pos_scores[:, 1:]); # [bs*M]
pn_prob = pu_prob.mean(dim=-1)/(1-tau) -  $\tau^+$  * pp_prob.mean(dim=-1)/(1-tau); #[bs, ]
dpl_loss = - log (pn_prob).mean();
根据 dpl_loss 相对于  $\Theta$  的梯度更新参数;
```

Result: 用户和物品表示 Θ 。

复杂度: 首先分析基线方法 BPR 的复杂度，该复杂度与评分函数 s 有关。这里，以维度 d 的矩阵分解作为示例，这个分析可以很容易地扩展到其他模型。给定一个批量大小为 bs 的 (u, i, j) 的训练三元组，前向评分计算涉及到总共 $2 \times bs$ 个得分预测，因此时间复杂度为 $\mathcal{O}(2bs \times d)$ 。在反向传播中，最多更新 $3 \times bs$ 个嵌入，涉及总共 $5bs \times d$ 次操作，因此 BPR 的时间复杂度为 $\mathcal{O}(bs \times d)$ 。类似地，对于 DPL，一个小批量数据涉及到总共 $(M + N + 1) \times bs$ 个评分计算和 $(M + N + 2) \times bs$ 个嵌入更新，涉及总共 $(2M + 2N + 3) \times bs \times d$ 个操作，时间复杂度为 $\mathcal{O}(bs \times d)$ ，因为在实践中通常将 M 和 N 设置为小的常数，例如 $M = 3$ 和 $N = 3$ 。特别地，当 $M = 0$ 且 $N = 1$ 时，DPL 涉及的操作数量与 BPR 相同。因此，相对于 BPR，DPL 具有严格的线性复杂度，没有任何 mini-batch 数据之外的计算或者存储开销。而前面的负采样算法，由于涉及计算用户每个训练三元组中用户 u 对应的评分向量，而推荐中，物品数量通常非常大，导致了时间复杂度的增加。

5.5 去偏成对损失的理论分析

DPL 的主要思想是改进使用正例-未标记数据对 (PU) 计算的偏倚概率 P_{PU} 。通过采样额外的正例和负例样本进行来估计式(5.17)所给出的理想优化目标“用户喜欢正例 \mathbf{x}_1 优于负例 \mathbf{x}_2 ”的概率 \hat{P}_{PN} , 然后用其替代原始的偏倚概率估计 P_{PU} 。为了证明 \hat{P}_{PN} 是一个良好的估计量, 首先证明 \hat{P}_{PN} 是 AUC 风险的无偏估计。

引理 5.1: 设正例 \mathbf{x}^+ 是从正例的类条件概率分布 $p^+(\mathbf{x})$ 独立采样的样本, 未标注样本 \mathbf{x} 是从边缘分布 $p(\mathbf{x}) = \tau^+ p^+(\mathbf{x}) + \tau^- p^-(\mathbf{x})$ 独立采样的样本。那么公式 (5.17) 给出的 \hat{P}_{PN} 是无偏的 AUC 风险估计:

$$\mathbb{E}\hat{P}_{\text{PN}} = R_{\text{AUC}}$$

证明:

$$\begin{aligned} \mathbb{E}\hat{P}_{\text{PN}} &= \int_{\mathbf{x}^+} \left[\frac{1}{N\tau^-} \sum_{\mathbf{x} \in \mathcal{D}^u} \mathbb{E}_{\mathbf{x} \sim p} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{M\tau^-} \sum_{\mathbf{x}' \in \mathcal{D}^+} \mathbb{E}_{\mathbf{x}' \sim p^+} h(\mathbf{x}^+, \mathbf{x}') \right] p^+(\mathbf{x}^+) d\mathbf{x}^+ \\ &= \int_{\mathbf{x}^+} \left[\frac{1}{\tau^-} \mathbb{E}_{\mathbf{x} \sim p} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{\tau^-} \mathbb{E}_{\mathbf{x}' \sim p^+} h(\mathbf{x}^+, \mathbf{x}') \right] p^+(\mathbf{x}^+) d\mathbf{x}^+. \end{aligned} \quad (5.20)$$

由于

$$\begin{aligned} &\frac{1}{\tau^-} \mathbb{E}_{\mathbf{x} \sim p} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{\tau^-} \mathbb{E}_{\mathbf{x}' \sim p^+} h(\mathbf{x}^+, \mathbf{x}') \\ &= \frac{1}{\tau^-} \int_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) p(\mathbf{x}) d\mathbf{x} - \frac{\tau^+}{\tau^-} \int_{\mathbf{x}'} h(\mathbf{x}^+, \mathbf{x}') p^+(\mathbf{x}') d\mathbf{x}' \\ &= \frac{1}{\tau^-} \int_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) [\tau^+ p^+(\mathbf{x}) + \tau^- p^-(\mathbf{x})] d\mathbf{x} - \frac{\tau^+}{\tau^-} \int_{\mathbf{x}'} h(\mathbf{x}^+, \mathbf{x}') p^+(\mathbf{x}') d\mathbf{x}' \end{aligned} \quad (5.22)$$

$$= \int_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) p^-(\mathbf{x}) d\mathbf{x} \quad (5.23)$$

$$= \int_{\mathbf{x}^-} h(\mathbf{x}^+, \mathbf{x}^-) p^-(\mathbf{x}^-) d\mathbf{x}^- \quad (5.24)$$

其中, 公式 (5.22) 是通过边缘分布的全概率分解 $p(\mathbf{x}) = \tau^+ p^+(\mathbf{x}) + \tau^- p^-(\mathbf{x})$ 得到, 公式 (5.24) 替换积分变量 \mathbf{x} 为 \mathbf{x}^- 以提高可读性。将公式 (5.24) 带入公式 (5.21), 有

$$\mathbb{E}\hat{P}_{\text{PN}} = \int_{\mathbf{x}^+} \int_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}^-) p^-(\mathbf{x}^-) p^+(\mathbf{x}^+) d\mathbf{x}^+ d\mathbf{x}^- \quad (5.25)$$

$$= R_{\text{AUC}}. \quad (5.26)$$

证毕。 \square

如果将方程(5.25)中的函数 h 替换为0-1损失 $\mathbb{I}(\mathbf{x}^+, \mathbf{x}^-)$, 那么方程(5.25)正好是AUC指标的严格定义。然而, 由于0-1损失函数是离散的, 因此在优化AUC指标时通常使用可微的替代损失函数。关于引理5.1的直观解释, 请参考图5.2。需要说明的是, 无偏估计的成立, 不需要 M, N 趋近于无穷大作为必要条件, 这得益于在成对学习的问题设置下, 一个正例-未标注样本对(PU)所对应的标签情况可以被枚举(参考图5.1)。

接下来, 本章寻求一个理想的优化目标以供DPL近似。

定义5.1: 对于固定的正例 \mathbf{x}^+ , 记 $\mathbb{P}_{\text{PN}} = \mathbb{E}_{\mathbf{x}^- \sim p^-} h(\mathbf{x}^+, \mathbf{x}^-)$ 为用户喜欢正例 \mathbf{x}^+ 强于负例 \mathbf{x}^- 的期望值。定义监督损失为这个对数似然对所有正例的期望值

$$\mathcal{L}_{\text{SUP}} = -\mathbb{E}_{\mathbf{x}^+ \sim p^+} \log \mathbb{P}_{\text{PN}} \quad (5.27)$$

最小化上述对数似然对所有正例样本的期望值, 将导致最大化任意正例样本优于任意真负例样本的似然, 这恰好是在完全监督数据下我们的优化目标。因此, \mathcal{L}_{SUP} 是DPL近似的理想目标。引理5.2证明了DPL估计量在 M 和 N 趋近无穷大时与该理想损失渐近一致。

引理5.2: 当 $M, N \rightarrow +\infty$ 时, 有

$$\mathcal{L}_{\text{DPL}} \rightarrow \mathcal{L}_{\text{SUP}} \quad (5.28)$$

证明: 勒贝格控制收敛定理^[123]表明, 对于一个有界的可测函数序列 f_n , 有

$$\lim_{n \rightarrow \infty} \int_{\Omega} f_n = \int_{\Omega} \lim_{n \rightarrow \infty} f_n$$

那么

$$\begin{aligned} \lim_{M, N \rightarrow +\infty} \mathcal{L}_{\text{DPL}} &= -\lim_{M, N \rightarrow +\infty} \mathbb{E}_{\mathbf{x}^+ \sim p^+} \log \left[\frac{1}{N\tau^-} \sum_{\mathbf{x} \in \mathcal{D}^u} h(\mathbf{x}^+, \mathbf{x}) \frac{\tau^+}{M\tau^-} \sum_{\mathbf{x}' \in \mathcal{D}^+} h(\mathbf{x}^+, \mathbf{x}') \right] \\ &= -\mathbb{E}_{\mathbf{x}^+ \sim p^+} \lim_{M, N \rightarrow +\infty} \log \left[\frac{1}{N\tau^-} \sum_{\mathbf{x} \in \mathcal{D}^u} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{M\tau^-} \sum_{\mathbf{x}' \in \mathcal{D}^+} h(\mathbf{x}^+, \mathbf{x}') \right] \end{aligned}$$

$$\begin{aligned}
 &= -\mathbb{E}_{\mathbf{x}^+ \sim p^+} \lim_{M,N \rightarrow +\infty} \log \left[\frac{1}{N\tau^-} \sum_{\mathbf{x} \in \mathcal{D}^u} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{M\tau^-} \sum_{\mathbf{x}' \in \mathcal{D}^+} h(\mathbf{x}^+, \mathbf{x}') \right] \\
 &= -\mathbb{E}_{\mathbf{x}^+ \sim p^+} \log \left[\frac{1}{\tau^-} \mathbb{E}_{\mathbf{x} \sim p} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{\tau^-} \mathbb{E}_{\mathbf{x}' \sim p^+} h(\mathbf{x}^+, \mathbf{x}') \right]
 \end{aligned} \tag{5.29}$$

应用公式(5.24)的结果，有

$$\begin{aligned}
 &\frac{1}{\tau^-} \mathbb{E}_{\mathbf{x} \sim p} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{\tau^-} \mathbb{E}_{\mathbf{x}' \sim p^+} h(\mathbf{x}^+, \mathbf{x}') \\
 &= \int_{\mathbf{x}^-} h(\mathbf{x}^+, \mathbf{x}^-) p^-(\mathbf{x}^-) d\mathbf{x}^- \\
 &= \mathbb{E}_{\mathbf{x}^- \sim p^-} h(\mathbf{x}^+, \mathbf{x}^-) \\
 &= \mathbb{P}_{\text{PN}}
 \end{aligned} \tag{5.30}$$

将公式(5.30)带入公式(5.29)，有

$$\mathcal{L}_{\text{DPL}} \rightarrow \mathcal{L}_{\text{SUP}},$$

证毕。 \square

引理 5.2 证明了当 M 和 N 趋近无穷大时，DPL 损失 \mathcal{L}_{DPL} 与理想的有监督损失 \mathcal{L}_{SUP} 是渐近一致的。然而，在实际应用中，只可能使用有限大小的 M 和 N ，导致了经验估计 $\hat{\mathcal{L}}_{\text{DPL}}$ 。接下来，引理 5.3 给出了经验估计 $\hat{\mathcal{L}}_{\text{DPL}}$ 的估计误差界 $|\hat{\mathcal{L}}_{\text{DPL}} - \mathcal{L}_{\text{SUP}}|$ 。

引理 5.3：以至少 $1 - \delta$ 的概率，如下不等式成立

$$|\hat{\mathcal{L}}_{\text{DPL}} - \mathcal{L}_{\text{SUP}}| \leq e^2 \sqrt{\frac{2\pi}{N}} + e^2 \tau^+ \sqrt{\frac{2\pi}{M}}$$

证明：不失一般性，证明使用余弦相似度作为得分函数来简化分析，这意味着所有嵌入被映射到半径为 1 的超球面上。 h 是一个将两个样本映射到一个 sigmoid 函数中的函数： $h(\mathbf{x}^+, \mathbf{x}) = \sigma(g(\mathbf{x}^+) - g(\mathbf{x}))$ 。由于 $g(\cdot) \in [0, 1]$ ，那么 $-2 \leq g(\mathbf{x}^+) - g(\mathbf{x}) \leq 2$ ，且 $\frac{1}{1+e^2} \leq h(\mathbf{x}^+, \mathbf{x}) \leq \frac{1}{1+e^{-2}}$ 。

对于固定正例 \mathbf{x}^+ ，记渐近优化目标和非渐近目标的被积函数之间的差值为 Δ ：

$$\Delta = |\log \left[\frac{1}{N\tau^-} \sum_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{M\tau^-} \sum_{\mathbf{x}'} p(\mathbf{x}^+, \mathbf{x}') \right] - \log \mathbb{E}_{\mathbf{x}^- \sim p^-} h(\mathbf{x}^+, \mathbf{x}^-)|$$

$$\begin{aligned}
 &= |\log[\frac{1}{N\tau^-} \sum_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{M\tau^-} \sum_{\mathbf{x}'} p(\mathbf{x}^+, \mathbf{x}')] - \log \mathbb{E}_{\substack{\mathbf{x} \sim p(\mathbf{x}) \\ \mathbf{x}' \sim p^+(\mathbf{x})}} [\frac{1}{\tau^-} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{\tau^-} h(\mathbf{x}^+, \mathbf{x}')]| \\
 &= |\log \frac{\frac{1}{N} \sum_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{M} \sum_{\mathbf{x}'} h(\mathbf{x}^+, \mathbf{x}')}{\mathbb{E}_{\substack{\mathbf{x} \sim p(\mathbf{x}) \\ \mathbf{x}' \sim p^+(\mathbf{x})}} [h(\mathbf{x}^+, \mathbf{x}) - \tau^+ h(\mathbf{x}^+, \mathbf{x}')] }|
 \end{aligned}$$

由于 $\mathbb{P}(|X| \geq \epsilon) = \mathbb{P}(X \geq \epsilon) + \mathbb{P}(-X \geq \epsilon)$, 因此

$$\mathbb{P}(\Delta \geq \epsilon) = \mathbf{I}(\epsilon) + \mathbf{II}(\epsilon) \quad (5.31)$$

其中

$$\mathbf{I}(\epsilon) = \mathbb{P}\left(\log \frac{\frac{1}{N} \sum_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{M} \sum_{\mathbf{x}'} h(\mathbf{x}^+, \mathbf{x}')}{\mathbb{E}_{\substack{\mathbf{x} \sim p(\mathbf{x}) \\ \mathbf{x}' \sim p^+(\mathbf{x})}} [h(\mathbf{x}^+, \mathbf{x}) - \tau^+ h(\mathbf{x}^+, \mathbf{x}')] } \geq \epsilon\right) \quad (5.32)$$

$$\leq \mathbb{P}\left(\frac{\frac{1}{N} \sum_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{M} \sum_{\mathbf{x}'} h(\mathbf{x}^+, \mathbf{x}')}{\mathbb{E}_{\substack{\mathbf{x} \sim p(\mathbf{x}) \\ \mathbf{x}' \sim p^+(\mathbf{x})}} [h(\mathbf{x}^+, \mathbf{x}) - \tau^+ h(\mathbf{x}^+, \mathbf{x}')] } - 1 \geq \epsilon\right) \quad (5.33)$$

$$\begin{aligned}
 &\leq \mathbb{P}\left(\frac{1}{N} \sum_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{M} \sum_{\mathbf{x}'} h(\mathbf{x}^+, \mathbf{x}') \right. \\
 &\quad \left. - \mathbb{E}_{\substack{\mathbf{x} \sim p(\mathbf{x}) \\ \mathbf{x}' \sim p^+(\mathbf{x})}} [h(\mathbf{x}^+, \mathbf{x}) - \tau^+ h(\mathbf{x}^+, \mathbf{x}')] \geq \frac{\epsilon}{1 + e^2}\right) \quad (5.34)
 \end{aligned}$$

式(5.33)是由于 $\log x \leq x - 1$ 对所有的 $x > 0$ 成立。式(5.34)是由于 $\mathbb{E}_{\substack{\mathbf{x} \sim p(\mathbf{x}) \\ \mathbf{x}' \sim p^+(\mathbf{x})}} [h(\mathbf{x}^+, \mathbf{x}) - \tau^+ h(\mathbf{x}^+, \mathbf{x}')] = (1 - \tau^+) \mathbb{E}_{\mathbf{x} \sim p^-(\mathbf{x})} h(\mathbf{x}^+, \mathbf{x}) \geq 1/(1 + e^2)$ 。第二项类似

$$\begin{aligned}
 \mathbf{II}(\epsilon) &= \mathbb{P}\left(\log \frac{\mathbb{E}_{\substack{\mathbf{x} \sim p(\mathbf{x}) \\ \mathbf{x}' \sim p^+(\mathbf{x})}} [h(\mathbf{x}^+, \mathbf{x}) - \tau^+ h(\mathbf{x}^+, \mathbf{x}')] }{\frac{1}{N} \sum_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{M} \sum_{\mathbf{x}'} h(\mathbf{x}^+, \mathbf{x}') } \geq \epsilon\right) \\
 &\leq \mathbb{P}\left(\frac{\mathbb{E}_{\substack{\mathbf{x} \sim p(\mathbf{x}) \\ \mathbf{x}' \sim p^+(\mathbf{x})}} [h(\mathbf{x}^+, \mathbf{x}) - \tau^+ h(\mathbf{x}^+, \mathbf{x}')] }{\frac{1}{N} \sum_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{M} \sum_{\mathbf{x}'} h(\mathbf{x}^+, \mathbf{x}') } - 1 \geq \epsilon\right) \\
 &\leq \mathbb{P}\left(\mathbb{E}_{\substack{\mathbf{x} \sim p(\mathbf{x}) \\ \mathbf{x}' \sim p^+(\mathbf{x})}} [h(\mathbf{x}^+, \mathbf{x}) - \tau^+ h(\mathbf{x}^+, \mathbf{x}')] \right. \\
 &\quad \left. - [\frac{1}{N} \sum_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{M} \sum_{\mathbf{x}'} h(\mathbf{x}^+, \mathbf{x}')] \geq \frac{\epsilon}{1 + e^2}\right) \quad (5.35)
 \end{aligned}$$

综合式(5.34)与式(5.35), 有

$$\mathbb{P}(\Delta \geq \epsilon) \leq \mathbb{P}\left(|\frac{1}{N} \sum_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) - \frac{\tau^+}{M} \sum_{\mathbf{x}'} h(\mathbf{x}^+, \mathbf{x}')|\right) \quad (5.36)$$

$$\begin{aligned}
 & -\mathbb{E}_{\substack{\mathbf{x} \sim p(\mathbf{x}) \\ \mathbf{x}' \sim p^+(\mathbf{x})}} [h(\mathbf{x}^+, \mathbf{x}) - \tau^+ h(\mathbf{x}^+, \mathbf{x}')] \geq \frac{\epsilon}{1 + e^2} \\
 = & \mathbb{P}\left(\left|\frac{1}{N} \sum_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) - \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} h(\mathbf{x}^+, \mathbf{x})\right| \right. \\
 & \left. - \left[\frac{\tau^+}{M} \sum_{\mathbf{x}'} h(\mathbf{x}^+, \mathbf{x}') - \mathbb{E}_{\mathbf{x}' \sim p^+(\mathbf{x})} \tau^+ h(\mathbf{x}^+, \mathbf{x}')\right] \geq \frac{\epsilon}{1 + e^2}\right) \quad (5.37)
 \end{aligned}$$

$$\begin{aligned}
 \leq & \mathbb{P}\left(\left|\frac{1}{N} \sum_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) - \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} h(\mathbf{x}^+, \mathbf{x})\right| \right. \\
 & \left. + \left|\frac{\tau^+}{M} \sum_{\mathbf{x}'} h(\mathbf{x}^+, \mathbf{x}') - \mathbb{E}_{\mathbf{x}' \sim p^+(\mathbf{x})} \tau^+ h(\mathbf{x}^+, \mathbf{x}')\right| \geq \frac{\epsilon}{1 + e^2}\right) \quad (5.38)
 \end{aligned}$$

$$\begin{aligned}
 \leq & \mathbf{III}(\epsilon) + \mathbf{IV}(\epsilon). \quad (5.39)
 \end{aligned}$$

其中

$$\mathbf{III}(\epsilon) = \mathbb{P}\left(\left|\frac{1}{N} \sum_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}) - \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} h(\mathbf{x}^+, \mathbf{x})\right| \geq \frac{\epsilon}{2(1 + e^2)}\right) \quad (5.40)$$

$$\mathbf{IV}(\epsilon) = \mathbb{P}\left(\left|\frac{\tau^+}{M} \sum_{\mathbf{x}'} h(\mathbf{x}^+, \mathbf{x}') - \mathbb{E}_{\mathbf{x}' \sim p^+(\mathbf{x})} \tau^+ h(\mathbf{x}^+, \mathbf{x}')\right| \geq \frac{\epsilon}{2(1 + e^2)}\right) \quad (5.41)$$

式 (5.38) 是由于 $|X - Y| \leq |X| + |Y|$ 成立, 式 (5.39) 由于 $\mathbb{P}(|X| + |Y| \leq \epsilon) \leq \mathbb{P}(|X| \leq \epsilon/2) + \mathbb{P}(|Y| \leq \epsilon/2)$ 。根据 McDiarmid's 不等式, 对于独立同分布的随机变量 X_1, X_2, \dots, X_n , 如果存在界 c_1, c_2, \dots, c_n 和函数 $f : \mathcal{X}_1 \times \mathcal{X}_2 \times \dots \times \mathcal{X}_n \rightarrow \mathbb{R}$, 使得 $|f(x_1, x_2, \dots, x_i, \dots, x_n) - f(x_1, x_2, \dots, x'_i, \dots, x_n)| \leq c_i$ 对所有 $i \in \{1, 2, \dots, n\}$ 成立, 那么, 对任意 $\epsilon > 0$, 有如下不等式

$$\mathbb{P}(|f(X_1, X_2, \dots, X_n) - \mathbb{E}[f(X_1, X_2, \dots, X_n)]| \geq \epsilon) \leq 2 \exp\left(-\frac{2\epsilon^2}{\sum_{i=1}^n c_i^2}\right).$$

记未标记样本的得分 $g(\mathbf{x})$ 是一个随机变量, 由于 $\frac{1}{1+e^2} \leq h(\mathbf{x}^+, \mathbf{x}) \leq \frac{1}{1+e^{-2}}$, 那么函数 $f : \mathcal{X}_1 \times \mathcal{X}_2 \times \dots \times \mathcal{X}_n \rightarrow \frac{1}{N} \sum_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x})$ 满足有界差异数性质, 边界为 $c_1 = c_2 = \dots = c_n = \frac{1}{N} \frac{e^2 - 1}{e^2 + 1}$, 于是

$$\mathbf{III}(\epsilon) \leq 2 \exp\left(-\frac{N\epsilon^2}{2(e^2 - 1)^2}\right) \quad (5.42)$$

$$\leq 2 \exp\left(-\frac{N\epsilon^2}{2e^4}\right). \quad (5.43)$$

记正样本的得分 $g(\mathbf{x}')$ 为随机变量, 函数 $f : \mathcal{X}_1 \times \mathcal{X}_2 \times \dots \times \mathcal{X}_m \rightarrow \frac{\tau^+}{M} \sum_{\mathbf{x}} h(\mathbf{x}^+, \mathbf{x}')$ 满

足偏微分有界性质，边界为 $c_1 = c_2 = \dots = c_m = \frac{\tau^+ e^2 - 1}{M e^2 + 1}$ ，于是

$$\mathbf{IV}(\epsilon) \leq 2 \exp\left(-\frac{M\epsilon^2}{2(e^2 - 1)^2\tau^{+2}}\right) \quad (5.44)$$

$$\leq 2 \exp\left(-\frac{M\epsilon^2}{2e^4\tau^{+2}}\right). \quad (5.45)$$

将式(5.43)和式(5.45)带回式(5.39)，有

$$\mathbb{P}(\Delta \geq \epsilon | \mathbf{x}^+) \leq 2 \exp\left(-\frac{N\epsilon^2}{2e^4}\right) + 2 \exp\left(-\frac{M\epsilon^2}{2e^4\tau^{+2}}\right). \quad (5.46)$$

为了获得 $|\hat{\mathcal{L}}_{\text{DPL}}(g) - \hat{\mathcal{L}}_{\text{DPL}}(g)|$ 的上界，跟随 DCL^[97] 的做法，通过应用 Jensen's 不等式，有

$$|\hat{\mathcal{L}}_{\text{DPL}} - \mathcal{L}_{\text{SUP}}| = \mathbb{E}_{\mathbf{x}^+} \log |\mathbf{I} - \mathbf{II}| \quad (5.47)$$

$$\leq \mathbb{E}_{\mathbf{x}^+} \Delta \quad (5.48)$$

$$= \mathbb{E}_{\mathbf{x}^+} [\mathbb{E}_\epsilon [\Delta | \mathbf{x}^+]] \quad (5.49)$$

$$= \mathbb{E}_{\mathbf{x}^+} \left[\int_0^{+\infty} \mathbb{P}(\Delta \geq \epsilon | \mathbf{x}^+) d\epsilon \right] \quad (5.50)$$

$$\leq \int_0^{+\infty} 2 \exp\left(-\frac{N\epsilon^2}{2e^4}\right) + 2 \exp\left(-\frac{M\epsilon^2}{2e^4\tau^{+2}}\right) d\epsilon \\ = e^2 \sqrt{\frac{2\pi}{N}} + e^2 \tau^+ \sqrt{\frac{2\pi}{M}}. \quad (5.51)$$

在式(5.50)中，外部的期望算子可以去掉，因为概率界对于所有固定的正例样本 \mathbf{x}^+ 都成立^[97]。证毕。 \square

5.6 实验评估

5.6.1 实验设置

5.6.1.1 数据集

本章在五个公开数据集上进行了实验：MovieLens-100k, MovieLens-1M, Yahoo!-R3, Yelp2018 和 Gowalla。前三个数据集包含用户评分，按照^[1,45,46]的方法将所有有评分的物品转换为隐式反馈。对于每个数据集，随机将其中 20% 的数据作为测试数据，剩下的 80% 用于训练。表 5.1 给出了数据集统计的摘要信息。

表 5.1 数据集统计

	用户数	物品数	总交互数	训练集交互数	测试集交互数	密度
MovieLens-100k	943	1,682	100,000	80k	20k	0.06304
MovieLens-1M	6,040	3,952	1,000,000	800k	200k	0.04189
Yahoo!-R3	5,400	1,000	182,000	146k	36k	0.03370
Yelp2018	31,668	38,048	1,561,406	1,249k	312k	0.00130
Gowalla	29,858	40,981	1,027,370	821k	205k	0.00084

5.6.1.2 评估指标

为了评估推荐的性能，我们采用常用的指标来评估 Top- K 推荐，这些指标包括精确度 (P)、召回率 (R) 和归一化折损累积增益 (NDCG)，其中 K 的取值为 5、10 和 20。由于这些指标的广泛使用，不在此处提供它们的定义。

5.6.1.3 实验设置

本章也使用两个推荐模型：经典的矩阵分解 (MF)^[124] 和较新的轻量级图卷积网络 (LightGCN)^[5]。前三个数据集的计算是在一台运行 Windows 10 操作系统的个人计算机上进行的，该计算机配备了 2.1 GHz 的 CPU、一张 RTX 1080Ti GPU 和 32 GB 的 RAM。后两个数据集的计算是在一台运行 Linux 操作系统的云服务器上进行的，该服务器配备了一颗 Xeon(R) Platinum 8358P CPU、一张 RTX A40 GPU 和 56GB 的 RAM。

5.6.1.4 对比算法

- BPR^[1]: 由于隐性反馈为 positive-unlabeled 数据，BPR 从用户进行交互的 (u, i) 对中采样正例，而负例则从用户未进行交互的未标记数据 (u, j) 中抽取。
- InfoNCE^[7]: InfoNCE 损失是机器学习中常用的损失函数，特别是在对比表示学习中。具体而言，InfoNCE 衡量了正样本 \mathbf{x}^+ 与一组负样本 $\mathbf{x}_i^-_{i=1}^N$ 之间的相似性，并应用了 softmax 函数：

$$\mathcal{L}_{\text{InfoNCE}} = -\mathbb{E}_{\substack{\mathbf{x}^+ \sim p^+(\mathbf{x}) \\ \mathbf{x}_i^- \sim p^-(\mathbf{x})}} \log \frac{\exp(g(\mathbf{x}^+))}{\exp(g(\mathbf{x}^+)) + \sum_{i=1}^N \exp(g(\mathbf{x}_i^-))}$$

InfoNCE 损失是噪声对比估计 (NCE) 以及 BPR 损失从一个负样本向 N 个负样本的推广。在实践中，由于负样本不可得，通常从未标记的样本中采样 \mathbf{x}_i^- 作

为负样本。

- DCL^[97]: 由于未标记数据中存在虚假负例, DCL 通过修正概率估计来进行虚假负例去偏。具体而言, DCL 提出了一个估计量来修正 $\mathcal{L}_{\text{InfoNCE}}$ 分母中的第二项

$$\mathcal{L}_{\text{DCL}} = -\mathbb{E}_{\substack{\mathbf{x}^+ \sim p^+(\mathbf{x}) \\ \mathbf{x}_i^- \sim p^-(\mathbf{x})}} \log \frac{\exp(g(\mathbf{x}^+))}{\exp(g(\mathbf{x}^+)) + Ng}$$

其中

$$g = \frac{1}{N\tau^-} \left(\sum_{i=1}^N \exp(g(\mathbf{x}_i)) - N\tau^+ \cdot \frac{\sum_{j=1}^K \exp(g(\mathbf{x}_j^+))}{K} \right)$$

g 估计量可以理解为真负样本分数的和。具体而言, $N\tau^+$ 伪负样本的数量的估计, 而 $\frac{\sum_{j=1}^K \exp(g(\mathbf{x}_j^+))}{K}$ 估计了 K 个伪负样本分数的均值。因此, 括号内的第二项对应于 N 个样本中所有伪负样本分数的和, 将其从 N 个随机选择的未标记样本的分数和 $\sum_{i=1}^N \exp(g(\mathbf{x}_i))$ 中减去, 则对应于 N 个样本中所有真负样本的分数和。

- HCL^[98]: 使用了 DCL 的去偏框架, 它还通过对每个随机选择的未标记样本进行加权来考虑困难样本挖掘, 具体如下所示

$$\omega_i^{\text{HCL}} = \frac{g(\mathbf{x}'_j)^\beta}{\frac{1}{N} \sum_{j=1}^N g(\mathbf{x}'_j)^\beta}. \quad (5.52)$$

其中, β 控制了挖掘难负例的程度。DCL 是 $\beta = 0$ 时 HCL 的一个特例。

5.6.2 实验结果

5.6.2.1 个性化推荐性能

从表 5.2 中可以得出第一个观察结果, 即 DPL 取得了最佳性能。与没有去偏机制的 BPR 和 InfoNCE 相比, DPL 显示出了显著的改进, 表明在隐性反馈数据中纠正偏倚的概率估计的必要性。与具有去偏机制的 DCL 和 HCL 相比, DPL 也取得了显著的改进, 主要是由于 DPL 在成对学习问题设置中的去偏机制的优势。在成对学习问题设置中, 未标记样本的数量为 1, 其对应的真实标签可以枚举 (参见图 5.1), 进而可以获得“用户更喜欢正例强于负例的概率”的无偏估计。然而, 在 N 个未标记样本的情况下, 根据二项式定理, 真实标签有 2^N 种可能的结果, 使得枚举每种情况变得困难。因此, DCL 和 HCL 的去偏机制主要依赖于数值逼近。

表 5.2 Top-k 推荐性能比较

数据集	模型	学习算法	Top-5 评估			Top-10 评估			Top-20 评估		
			Precision	Recall	NDCG	Precision	Recall	NDCG	Precision	Recall	NDCG
MovieLens-100k	MF	BPR	0.3900	0.1301	0.4143	0.3363	0.2164	0.3967	0.2724	0.3298	0.3962
		InfoNCE	0.4081	0.1388	0.4324	0.3452	0.2266	0.4095	0.2793	0.3497	0.4118
		DCL	0.4168	0.1434	0.4458	0.3513	0.2291	0.4202	0.2835	0.3546	0.4207
		HCL	0.4263	0.1463	0.4539	0.3565	0.2323	0.426	0.2849	0.3564	0.4242
		DPL(Proposed)	0.4348	0.1523	0.4643	0.3635	0.2379	0.4356	0.2914	0.3588	0.4338
MovieLens-1M	MF	BPR	0.3944	0.1231	0.4204	0.3346	0.2189	0.4017	0.2658	0.3281	0.3986
		Info_NCE	0.3924	0.1343	0.4209	0.3349	0.2183	0.4006	0.2679	0.3289	0.3976
		DCL	0.3962	0.1367	0.4243	0.3361	0.2194	0.4022	0.2695	0.3329	0.4006
		HCL	0.4197	0.1461	0.4501	0.3458	0.2256	0.4188	0.2802	0.3446	0.4182
		DPL(proposed)	0.4333	0.1486	0.4627	0.3596	0.2344	0.4324	0.2919	0.3585	0.4331
Yahoo!-R3	MF	BPR	0.3843	0.0855	0.4027	0.3353	0.1430	0.3737	0.2798	0.2244	0.3572
		InfoNCE	0.3820	0.0879	0.4003	0.3339	0.1478	0.3728	0.2821	0.2358	0.3605
		DCL	0.4009	0.0934	0.4209	0.3472	0.1546	0.3894	0.289	0.2423	0.3731
		HCL	0.4112	0.0969	0.4317	0.3552	0.1585	0.3991	0.2959	0.2475	0.3825
		DPL(proposed)	0.4212	0.0998	0.4407	0.3624	0.1625	0.4071	0.2991	0.2518	0.3891
Yelp2018	MF	BPR	0.4095	0.0953	0.4305	0.3512	0.1547	0.3985	0.2915	0.2405	0.3781
		InfoNCE	0.4121	0.0986	0.4386	0.359	0.1594	0.4041	0.2979	0.2482	0.3869
		DCL	0.4104	0.0982	0.4291	0.3544	0.1597	0.3977	0.2965	0.2511	0.3842
		HCL	0.4107	0.0948	0.4300	0.3514	0.1542	0.3950	0.2916	0.2413	0.3775
		DPL(proposed)	0.4217	0.1003	0.4429	0.3620	0.1625	0.1866	0.2989	0.2511	0.3896
Gowalla	MF	BPR	0.1417	0.1052	0.1587	0.1064	0.1573	0.1641	0.0768	0.2259	0.1913
		Info_NCE	0.1429	0.1065	0.1615	0.1080	0.1601	0.1664	0.0786	0.2316	0.1952
		DCL	0.1454	0.1083	0.1635	0.1091	0.1618	0.1692	0.079	0.2327	0.1974
		HCL	0.1460	0.1097	0.1638	0.1096	0.1628	0.1697	0.0792	0.2336	0.1976
		DPL(proposed)	0.1491	0.1091	0.1652	0.1108	0.1641	0.1712	0.0801	0.2351	0.2012
LightGCN	LightGCN	BPR	0.1479	0.1101	0.1693	0.1126	0.1669	0.1760	0.0814	0.2389	0.2047
		Info_NCE	0.1417	0.1074	0.1676	0.1099	0.1633	0.1719	0.0798	0.2354	0.2007
		DCL	0.1456	0.1092	0.1642	0.1089	0.1622	0.1697	0.079	0.2333	0.1982
		HCL	0.1412	0.1139	0.1718	0.113	0.1683	0.1776	0.0812	0.2394	0.2059
		DPL(proposed)	0.1504	0.1111	0.1697	0.1131	0.1670	0.1757	0.0825	0.2412	0.2054
LightGCN	MF	BPR	0.0398	0.0228	0.0435	0.0339	0.0389	0.0456	0.0284	0.065	0.0538
		Info_NCE	0.0429	0.0246	0.047	0.0365	0.0417	0.0491	0.0305	0.07	0.058
		DCL	0.0486	0.0278	0.0531	0.0410	0.0466	0.0552	0.0342	0.0777	0.0648
		HCL	0.0515	0.0305	0.0566	0.0459	0.0541	0.0622	0.0383	0.0894	0.0736
		DPL(proposed)	0.0543	0.0325	0.0595	0.0463	0.0551	0.0630	0.0389	0.0914	0.0749
LightGCN	Yelp2018	BPR	0.0556	0.0330	0.0610	0.0473	0.0560	0.0644	0.0391	0.0914	0.0757
		Info_NCE	0.0553	0.0329	0.0607	0.0473	0.0558	0.0642	0.0390	0.0911	0.0754
		DCL	0.0559	0.0331	0.0612	0.0472	0.0557	0.0642	0.0391	0.0914	0.0756
		HCL	0.0563	0.0335	0.0617	0.0477	0.0564	0.0648	0.0393	0.0920	0.0760
		DPL(proposed)	0.0604	0.0364	0.0657	0.0513	0.0615	0.0696	0.0423	0.1003	0.0821
LightGCN	MF	BPR	0.0728	0.0748	0.1000	0.0555	0.1116	0.1063	0.0414	0.1625	0.1209
		Info_NCE	0.0739	0.0757	0.1016	0.0560	0.1122	0.1076	0.0422	0.1650	0.1230
		DCL	0.0746	0.0769	0.1023	0.0568	0.1147	0.1088	0.0426	0.1664	0.1238
		HCL	0.0755	0.0774	0.1035	0.0574	0.1151	0.1098	0.0432	0.1693	0.1256
		DPL(proposed)	0.0815	0.0827	0.1100	0.0628	0.1243	0.1174	0.0473	0.1815	0.1340
LightGCN	Gowalla	BPR	0.0735	0.0753	0.1007	0.0560	0.1119	0.1069	0.0419	0.1641	0.1218
		Info_NCE	0.0743	0.0760	0.1022	0.0566	0.1132	0.1084	0.0423	0.1649	0.1231
		DCL	0.0748	0.0763	0.1027	0.0569	0.1132	0.1088	0.0424	0.1656	0.1236
		HCL	0.0794	0.0804	0.1084	0.0608	0.1199	0.1147	0.0453	0.1740	0.1319
		DPL(proposed)	0.0867	0.0891	0.1164	0.0662	0.1329	0.1242	0.0494	0.1936	0.1417

第二个观察结果是，作为 InfoNCE 的一个特例，BPR 损失在性能上稍逊于具有 N 个负样本的 InfoNCE，特别是在大型稀疏数据集上。这是因为较大的负样本数量 N 会更紧密地推高互信息下界^[7]。然而，在 PU 数据集中，较大的 N 并不总是更好，因为较大的 N 通常会导致对难样本（即在嵌入空间中与锚点（即用户嵌入）更接近的样本）赋予更大的梯度值。如果这个样本是一个伪负例，较大的梯度值会严重损害模型的性能。

第三个观察结果是，具有去偏机制的 DCL、HCL 和 DPL 通常优于缺乏去偏机制的 BPR 和 InfoNCE，尤其是在具有较高正类先验的数据集上。此外，HCL 通过将高分数的难样本分配更高的权重，给负样本分配较大的梯度值，在去偏的基础上通过隐式难负样本挖掘取得良好的结果。然而，应该谨慎调整相应的困难样本挖掘参数 β ，以防止伪负例对模型性能的伤害。这是因为如果困难样本是一个伪负例，模型性能将受到损害，但如果是一个真负例，模型性能将从难样本中受益。这种现象在协同过滤中被称为“探索与利用的权衡”^[48]，在计算机视觉中被称为“均匀性与对齐性的困境”^[125]。

5.6.2.2 超参数分析

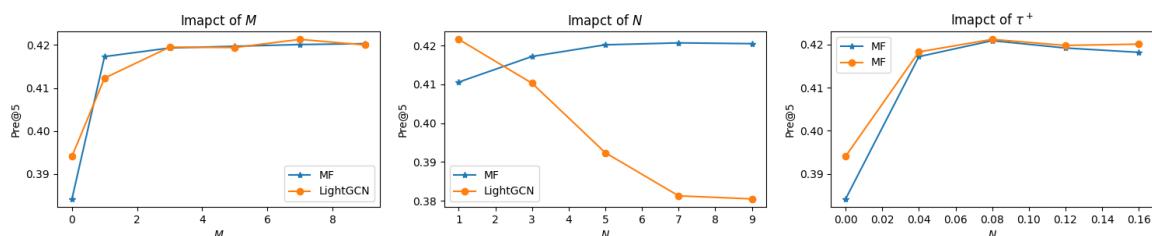


图 5.3 不同超参数的影响

M 参数的影响：参数 M 控制用于纠正采样偏差的额外正例数量。当 $M=0$ 时，没有去偏机制。对于 MF 模型，较大的 M 和 N 通常会导致性能提升，从 $M=0$ 到 $M=1$ 观察到显著的性能提升，突出了使用额外正例进行去偏的重要性。然而，当 M 和 N 超过一个小的常数时，性能不再提高。这是因为进一步增加 M 和 N 所带来的估计准确性的边际收益变得有限，正如之前的理论分析所述。

N 参数的影响：参数 N 控制用于计算 PU 概率的负例数量。与 MF 模型类似，增加 M 始终会提高 LightGCN 的性能。然而，当负例数量 N 增加时，模型性能意外地

下降。这个非预期的结果可能应该归因于较大的 N 会降低难负例样本的梯度值。根据詹森不等式， $-\log(\frac{1}{N} \sum_{i=1}^N P_n) \leq -\frac{1}{N} \sum_{i=1}^N \log P_n$ ；因此，过高的 N 值削弱了难负样本对学习算法的贡献。理论分析假设得分 $g(\cdot)$ 是独立同分布 (i.i.d.) 变量，并且估计误差随着 N 的增加而减小。然而，基于图神经网络编码器的聚合机制严重影响了 $g(\cdot)$ 值的独立同分布性质。因此，当推荐模型为难以优化的神经网络模型如 LightGCN 时，应当设置较小的 N 值。

正类先验 τ^+ 的影响：随着 τ^+ 增加，MF 和 LightGCN 模型的性能呈现出倒 U 型曲线，先增加后减少。这是因为设置过低或过高的 τ^+ 值可能导致公式(5.17)的估计结果偏倚。一种常见的设置 τ^+ 值的方法是将观察到的正交互作用的数量 $|\mathcal{D}^+|$ 视为伯努利试验的结果，该试验总共发生 $|\mathcal{U}| \times |\mathcal{I}|$ 次，成功的次数为观察到的交互作用数量。数据集的密度 $|\mathcal{D}^+| / (|\mathcal{U}| \times |\mathcal{I}|)$ 可以作为设置 τ^+ 值的参考。然而，需要注意的是，基于数据集密度设置的 τ^+ 值也是一种偏倚估计，低估了真实值，因为数据中的所有未观察到的 (u, i) 对都被视为负样本，导致成功次数的低估。详细讨论可参考 [126,127]。

5.6.2.3 估计校正 vs 负采样

负采样和估计校正代表了解决采样偏差问题的两种不同技术方法。负采样的核心思想是在模型外选择理想的样本并将其喂入模型训练，而估计校正的思想是使用随机样本，但对损失函数进行修正。从贝叶斯统计的角度来看，负采样利用了两种类型的信息：先验信息，如物品类别和流行度，这是静态的信息，用于采样用户不喜欢的负样本；样本信息，如分数和排名位置，这是动态的信息，在模型训练过程中不断调整，并用于采样嵌入接近锚点嵌入（得分较高）的难样本。各种负采样算法之间的差异在于它们如何利用和处理这两种类型的信息。上一章的贝叶斯负采样算法制定了后验概率意义上的负信号测度，并提出了理论上最优的采样规则。表 5.3 中比较了 DPL 和 BNS 的性能。

性能比较：DPL 在 MovieLens100k 数据集上取得了更好的性能，并且在 MovieLens1M 数据集上与 BNS 表现相似，这表明基于校正估计的 DPL 作为一种替代负采样算法是可行的。此外，我们发现使用困难、得分较高的未标记样本来训练 DPL 可以带来一些性能改进。当使用困难样本训练 DPL 时，需要设置相对较大的 τ^+ 值。在表 5.3 中，分别将 τ^+ 设置为 0.3 和 0.25，这比数据集本身的密度要高得多。这是因为

表 5.3 去偏成对学习算法与负采样算法的性能比较

数据集	模型	方法	Top-5			Top-10			Top-20		
			Precision	Recall	NDCG	Precision	Recall	NDCG	Precision	Recall	NDCG
MovieLens-100k	MF	BPR	0.3900	0.1301	0.4143	0.3363	0.2164	0.3967	0.2724	0.3298	0.3962
		BNS	0.4205	0.1467	0.4558	0.3463	0.2290	0.4217	0.2762	0.3466	0.4176
		DPL(Proposed)	0.4348	0.1523	0.4643	0.3635	0.2379	0.4356	0.2914	0.3588	0.4338
		DPL with Hard Samples	0.4401	0.1579	0.4692	0.3713	0.2407	0.4395	0.2940	0.3592	0.4351
MovieLens-1M	MF	BPR	0.3843	0.0855	0.4027	0.3353	0.1430	0.3737	0.2798	0.2244	0.3572
		BNS	0.4207	0.1062	0.4324	0.3518	0.1703	0.4191	0.3045	0.2614	0.4002
		DPL(proposed)	0.4212	0.0998	0.4407	0.3624	0.1625	0.4071	0.2991	0.2518	0.3891
		DPL with Hard Samples	0.4251	0.1012	0.4412	0.3649	0.1701	0.4151	0.3012	0.2539	0.3922

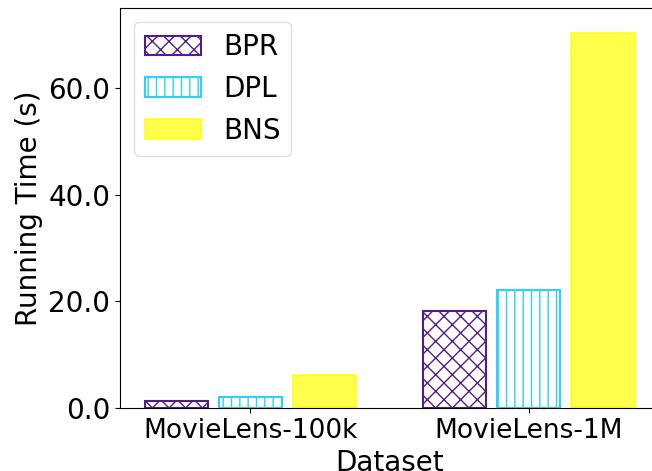


图 5.4 去偏成对学习算法和负采样算法的运行时间比较

使用困难样本训练 DPL 会增加模型遇到误分类负样本的概率，这相当于人为地改变了喂入模型的训练样本的正类先验。

运行时间：运行时间是在一台配备 2.1 GHz CPU、RTX 1080Ti GPU 和 32 GB RAM 的个人计算机上进行测试的。图 5.4 显示了三种算法在五个数据集上进行一次 epoch 训练的运行时间，其中协同过滤模型固定为 MF，批大小固定为 1024。在计算经验分布函数时，BNS 只采用 mini-batch 内的样本近似计算。如图 5.4 所示，DPL 的实际运行时间仅略长于 BPR，这与之前的时间复杂性分析一致。即使采用批内样本近似以节省计算成本，BNS 的运行时间仍然是 BPR 和 DPL 的 3-5 倍。这是因为动态负采样需要预测得分来指导负采样，但基于 GPU 的批计算在进行前向传播以预测得分之前需要固定的负样本。因此，动态负采样通常通过将额外的负样本作为候选项加载到小批量数据中来实现，从而导致额外的计算成本。此外，一些最先进的动态

负采样算法在前几个训练 epoch 中需要样本的排名位置或预测得分的方差^[128]，这要求模型在进行前向传播时计算整个用户-物品评分矩阵的预测得分，而不仅仅是小批量数据内的得分，从而导致指数级的时间复杂度。

总之，在具有丰富的辅助信息的场景中，建议使用可以灵活组合先验信息和模型信息的负采样算法。在没有可用的辅助信息进行监督的场景中，建议使用 DPL 方法进行损失修正。

5.7 本章小结

本章专注于解决来自正样本-未标记隐性反馈数据的采样偏差问题，但采用了与显式的负采样不同的技术方法。具体而言，本章提出了一种用于隐性反馈采样偏差的估计校正方法，从而为成对学习提供了一种修正的损失函数，称为去偏对数损失(DPL)。DPL 的关键思想是纠正由伪负例导致的概率估计偏差，从而修正梯度以近似完全监督数据的梯度。所提出的目标函数易于实现，不需要额外的辅助信息进行监督，也不需要过多的存储和计算开销。在保持严格的相对于 BPR 的线性时间复杂度情况下，DPL 取得了相对于负采样相当或更优的表现。

式(5.2)展示了成对损失和对比损失的联系，成对损失函数是负例个数 $N=1$ 的对比损失的特例。本章聚焦于成对损失的去偏研究，DPL 的去偏机制的核心可以参考图 5.1，在只有一个负例的情况下，未标注样本的真实标签有且只有两种可能性，很容易被枚举出来。在自监督对比学习中，一个广泛的共识是负例个数 N 越大，学到的对比表示在下游任务中泛化表现越好^[7,97,98]，这是由于更大的 N 推高了已知数据和待预测数据互信息的下界。但是，随着未标注样本 N 的增长，真实的标签的可能性指数级增长为 2^N 种，难以被枚举出来，导致本章的去偏思路则难以向更大的负例数推广。那么如何将去偏差的方法向更一般的对比损失推广？此外，对于在自监督对比学习任务而言，除了去偏以防止伪负例被推远，还有一个重要的硬负例挖掘任务，使真负例被推远。本章只考虑到前者。如何同时兼顾伪负例去偏和硬负例挖掘任务？下一个章节将针对这两个问题给出解决方案。

6 对比损失函数校正研究

在对比学习中，负例个数 N 推高了已知数据和待预测数据互信息的下界，使得学到的对比表示在下游任务中泛化表现越好^[7,97,98]。然而随着未标注样本 N 的增长，真实的标签的可能性指数级增长为 2^N 种，难以被枚举出来，导致上一章的去偏思路难以向更大的负例数推广；此外，对于在自监督对比学习任务而言，一方面要考虑伪负例去偏以防止伪负例被推远，另一方面还要考虑硬负例挖掘任务，使真负例被推远。本章聚焦于负例个数大于 1 的更一般的自监督对比学习的去偏差任务，并且进一步将硬负例挖掘任务纳入考量，通过引入重要性权重校正未标注样本的权重，得到了一种校正后的对比损失函数，称为贝叶斯自监督对比学习损失（Bayesian self-supervised contrastive loss, BCL）。从损失函数的数值来看，BCL 提供了完全监督数据下对比损失函数的渐进一致估计；从单个重要性权重来看，BCL 提供了未标注样本是真负样本的后验概率估计。在数值实验，四个公图像数据集，以及五个推荐数据集进行的实验研究验证了所提出的学习方法的有效性。

6.1 引言

无监督学习因为不依赖人工标注数据而广受关注。然而，在没有监督信号的情况下学习，如何到良好的表示一直是机器学习领域面临的一个长期挑战。最近，对比学习（contrastive learning）作为一种有希望的解决方案被提出，通过利用对比损失（contrastive loss）来训练表示编码器，从而改善无监督学习的表示能力。对比学习已经在各个领域展现出显著的成功，并且在计算机视觉、自然语言处理和推荐系统等任务中得到广泛应用，甚至在部分任务上超过了传统的有监督学习方法。

对比学习采用从比较种学习（learn-to-compare）的范式^[24]，通过区分观察数据和噪声数据来使模型免于重建数据的像素级信息^[7]。虽然表示编码器 f 和相似性度量会因任务而异^[9,29,30]，但它们都共享一个将相似对 (x, x^+) 和不相似对 (x, x^-) 进行对比的基本思想，通过优化对比损失^[32]来训练 f ，如 NCE 损失^[24]，InfoNCE 损失^[7]，Infomax 损失^[33]，渐进对比损失^[32]等。这些损失函数隐式^[7]或显式^[33]地优化了已知数据和待预测数据的互信息下界。Arora 等人^[34]为分类任务的对比学习的

泛化界限提供了理论分析。在不同领域的许多应用中都观察到了监督对比学习的显著成功^[35,36]，但其主要局限性来自于对手动标注数据集的依赖^[26]。自监督对比学习^[8,9,35,129]可以看作无监督学习的一个分支，它能够在无需人工标注数据的情况下学习表示，并且利益几乎所有类型的下游任务^[26,37-41]。

然而，当前的自监督对比学习仍存在一些挑战和局限性，主要还是由于无监督或者弱监督引发的 positive-unlabeled 问题。通常的做法是通过在锚点 x 上进行某些语义不变的数据增强^[8]来获得正样本 x^+ ，如随机裁剪和翻转^[7]，图像旋转^[76]，和颜色失真^[77]等；而负样本 x^- 则简单地从未标签数据中抽取，从而构成了 positive-unlabeled 问题。特别是，自监督对比学习面临两个冲突的任务^[26,98]：伪负样本的偏差修正和真负样本的硬负例挖掘。本章聚焦于更一般的自监督对比学习的去偏差任务，并且综合考虑了去偏任务和硬负例挖掘任务，称为贝叶斯自监督对比学习损失（Bayesian self-supervised contrastive loss, BCL）。BCL 利用重要性权重来纠正使用从未标记数据中随机抽取的负样本所带来的偏差，提供了完全监督数据下对比损失函数的渐进一致估计，以及未标注样本是真负样本的后验概率估计，为去偏任务和硬负例挖掘任务提供了灵活统一的框架。

6.2 对比损失优化目标分析

CPC^[7]提出了在自监督对比学习中广泛使用的 InfoNCE 损失，并构建了 InfoNCE 损失的优化与互信息优化的联系。在理想的监督信息下，优化 InfoNCE 损失会导致优化已知数据和待预测数据的互信息下界，免于迫使编码器重建样本的像素级信息。然而，在负例未标注的情况下，本节将说明优化 InfoNCE 损失会导致另外一种不同的结果，即，互信息下界的不可控优化。本小节的内容主要是分析负例未标注时，有偏的对比损失与互信息优化的理论关联，阅读与否不影响对后面的算法内容理解。

在有监督对比学习（supervised contrastive learning）的背景下，可以通过随机抽取与锚点不同类的真负例 $x^- \sim p^-$ 来轻松构建不相似的样本对 (x, x^-) ，通过优化如下有监督的损失

$$\mathcal{L}_{\text{SUP}} = \mathbb{E}_{\substack{x \sim p_d \\ x^+ \sim p^+ \\ x^- \sim p^-}} \left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \sum_{i=1}^N e^{f(x)^T f(x_i^-)}} \right] \quad (6.1)$$

以训练一个编码器 $f : \mathcal{X} \rightarrow \mathbb{R}^d/t$ ，该编码器将数据点 x 映射到半径为 $1/t$ 的超球面

\mathbb{R}^d 上，其中 t 是温度缩放参数。本章中，在理论分析中也将 t 设为 1 以简化分析。

然而，在自监督对比学习（self-supervised contrastive learning）的场景下，由于样本的标签不可得，无法获得真负例的类条件概率分布 $x^- \sim p^-$ ，标准方法是从数据分布 p_d 中抽取 N 个样本，这些样本被假设为 x 的负样本，以优化以下的 InfoNCE 损失：

$$\mathcal{L}_{\text{BIASED}} = \mathbb{E}_{\substack{x \sim p_d \\ x^+ \sim p^+ \\ x' \sim p_d}} \left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \sum_{i=1}^N e^{f(x)^T f(x'_i)}} \right]. \quad (6.2)$$

这也被称为“有偏对比损失”^[97]，因为从 p_d 中抽取的那些随机未标注样本 x' ，它是伪负例（正例）的概率为 τ^+ 。

依照前文的符号标记，记 $x' \in \text{TN}$ 表示 x^- 是相对于锚点数据 x 的真负例（TN）样本，记 $x' \in \text{FN}$ 表示 x' 是相对于锚点 x 的伪负例（FN）样本，记 $\hat{x}^+ = e^{f(x)^T f(x^+)}$ ，表示正例和锚点的相似度得分。注意， x' 是 TN 还是 FN 是针对特定的锚点 x 而言的，在接下来的讨论中，为简洁起见，本章省略了正负样本是“相对于锚点 x ”的概念表述。已经证明，对于 $\{x'_i \in \text{TN}\}_{i=1}^N$ ，优化完全监督的 InfoNCE 损失（式(6.1)）将导致正例的相似度分数正比于密度比 $\frac{p^+}{p^-}$ ^[7,130]

$$\hat{x}^+ \propto p^+ / p^-. \quad (6.3)$$

互信息的最大化是科学和工程中的一个基本问题^[130,131]，密度比 p^+ / p^- 保存了未来信息和当前信号的互信息（MI）。为了使式 (6.1) 最小化，模型优化使得正例的相似度得分增加，从而使得密度比 p^+ / p^- 增加，间接地实现最大化互信息。

现在考虑式 (6.2) 中从未标注样本中采样的 InfoNCE 损失 $\mathcal{L}_{\text{BIASED}}$ ，它的含义是从 N 个未标记样本中分类出一个正样本的交叉熵。为了便于分析，将 x^+ 记为 x_0 。给定 $N + 1$ 个样本，可以通过以下方式推导出一个数据点 x_0 是正样本的后验概率：

$$\begin{aligned} P(x_0 \in \text{pos} | \{x_i\}_{i=0}^N) &= \frac{p^+(x_0) \prod_{i=1}^N p_d(x_i)}{\sum_{j=0}^N p^+(x_j) \prod_{i \neq j} p_d(x_i)} \\ &= \frac{p^+(x_0) / p_d(x_0)}{p^+(x_0) / p_d(x_0) + \sum_{j=1}^N p^+(x_j) / p_d(x_j)} \end{aligned} \quad (6.4)$$

最小的损失值 $\mathcal{L}_{\text{BIASED}}$ 在后验概率为 1 时取到，通过密度比取如下极限实现 $p^+(x_0) / p_d(x_0) \rightarrow +\infty$ 或 $p^+(x_j) / p_d(x_j) \rightarrow 0$ 。于是，最小化 $\mathcal{L}_{\text{BIASED}}$ 导致正例分数正比

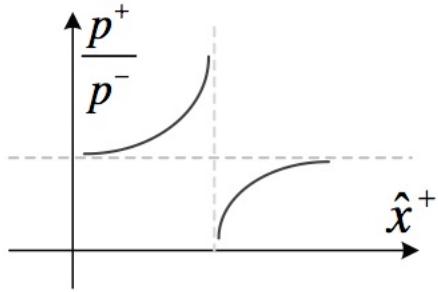


图 6.1 有偏对比损失与互信息优化示意图

于正例和未标注数据的密度比

$$\hat{x}^+ \propto p^+/p_d. \quad (6.5)$$

需要注意的是，这与公式(6.3)是不同的，前者是正例与负例的密度比，后者是正例与未标注样本的密度比。记 $\hat{x}^+ = m \cdot p^+/p_d, m \geq 0$ 。下面探讨优化 \hat{x}^+ 和优化目标 p^+/p^- 之间的差距。将全概率公式 $p_d = p^- \tau^- + p^+ \tau^+$ 代入公式(6.5)，有

$$\hat{x}^+ = m \cdot \frac{p^+}{p^- \tau^- + p^+ \tau^+}. \quad (6.6)$$

整理上式，有

$$p^+/p^- = \frac{\hat{x}^+ \cdot \tau^-}{m - \hat{x}^+ \cdot \tau^+}. \quad (6.7)$$

图 6.1 展示了公式(6.7)作为分数函数的近似形状，揭示了使用 PU 数据时，InfoNCE $\mathcal{L}_{\text{BIASED}}$ 损失优化和互信息(MI)优化之间的不一致性。也就是说，在使用 PU 数据优化 InfoNCE 损失时，增加 \hat{x}^+ 并不会导致 p^+/p^- 的单调增加。跳跃间断点的存在表明， $\mathcal{L}_{\text{BIASED}}$ 的优化不一定会导致可控的 MI 优化。MI 优化的难题源于以下事实：并非所有的 $\{x'_i\}_{i=1}^N$ 都是 T_N 样本，因为它们是从数据分布 p_d 中随机抽取的。这导致在分解数据分布 p_d 时，公式(6.6)的分母中包含了 p^+ 。图 6.2 提供了直观的解释。这五个随机抽样的样本实际上包含一个伪负例 F_N 样本。这样的伪负样本应该被拉近到锚点 x 附近。然而，由于它被错误地当作负样本时，在模型训练过程中，它将被推离锚点，这破坏了嵌入表示的语义结构^[125]。

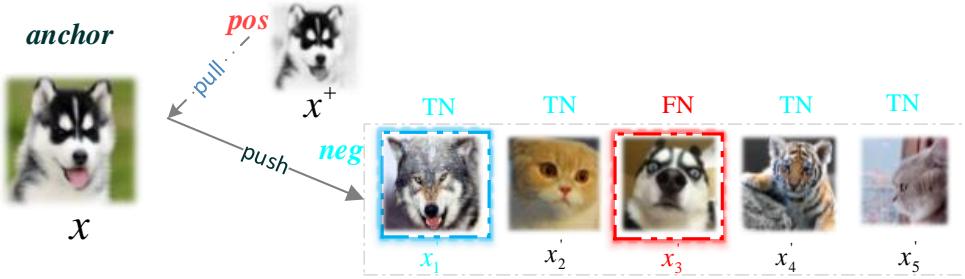


图 6.2 自监督对比学习的伪负例去偏和硬负例挖掘示意图

6.3 贝叶斯自监督对比学习算法

6.3.1 伪负例去偏与硬负例挖掘

本小节使用图像的例子来阐释自监督对比学习中的最重要两个任务：伪负例去偏和硬负例挖掘。对于一个锚点为“狗”的图像，正例通过语义不变的图像增强获得，负例从未标记图像中随机抽样，假设共得到五个随机负样本 $\{x'_1, x'_2, x'_3, x'_4, x'_5\}$ 。其中 x'_3 的真实标签是“狗”，是与锚点同类的伪负例 (FN)。剩余 4 个随机负样本是与锚点不同类的真负例 (TN)，其中 x'_1 的真实标签与锚点不同类的“狼”，但是它与锚点很相似难以区分，称为困难负样本 (hard negative sample)。

对比学习的任务是拉近正样本，推远负样本。由于随机负样本可能是真负例或者伪负例，那么根据未标注样本的标签不同，就存在两个不同的考量：

- 伪负例去偏：对于与锚点同类的伪负例 x'_3 ，应当防止它被推离锚点，保持“狗”这个类的样本距离比较近，也称为对齐 (alignment)。
- 硬负例挖掘：对于与锚点不同类的真负例 x'_1 ，应当把它推远离锚点，保持“狗”和“狼”这两个类之间的样本距离比较远，也称为均匀 (uniformity)。

这个例子也可以类比在推荐系统中，略微的差异是锚点的选择，图像中任意样本都可以做锚点，而在推荐系统中通常只有用户才会作为锚点。推荐系统中的正例是从观测到的交互中获得，负例是从未交互的物品中随机抽样获得。根据未标注样本标签的不同，同样也面临着上述硬负例挖掘和伪负例去偏的任务。

考虑 $\{x'_1, x'_2, x'_3, x'_4, x'_5\}$ 的标签，每个标签只有两种可能：与锚点类型不同时为真负例，与锚点类型相同时为伪负例。根据二项式定理，五个未标注样本的真实标签共有 $2^5 = 32$ 种可能的情况。当 N 很大时，枚举每一种情况，并对每一种情况进行

行极大似然估计是非常困难的，上一章的思想难以向较大负例个数 N 推广。受到经典的重要性采样 (importance sampling) 技术的启发，可以在每个未标注样本 x'_i 添加一个重要性权重 ω_i ，以纠正未标记样本中抽取负样本所产生的偏差。那么包含重要性权重的对比损失函数可以重写为

$$\begin{aligned}\mathcal{L} &= \mathbb{E}_{\substack{x \sim p_d \\ x^+ \sim p^+ \\ x' \sim p_d}} \left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \sum_{i=1}^N \omega_i \cdot e^{f(x)^T f(x'_i)}} \right] \\ &= \mathbb{E}_{\substack{x \sim p_d \\ x^+ \sim p^+ \\ x' \sim p_d}} \left[-\log \frac{\hat{x}^+}{\hat{x}^+ + \sum_{i=1}^N \omega_i \cdot \hat{x}_i} \right]\end{aligned}\quad (6.8)$$

式 (6.8) 是把正例与锚点的相似度得分 $e^{f(x)^T f(x^+)}$ 记为 \hat{x}^+ ，把负例 (实际上是未标注样本) 与锚点的相似度得分 $e^{f(x)^T f(x')}$ 记为 \hat{x} ，以保持分析过程中符号的简洁性。

下面考察在包含重要性权重后，权重 ω_i 是如何控制未标注样本 x'_i 对模型的参数学习与更新。损失函数相对于模型的参数 Θ 的梯度可以计算为：

$$\frac{\partial \mathcal{L}}{\partial \Theta} = \frac{\partial \mathcal{L}}{\partial \hat{x}'_i} \cdot \frac{\partial \hat{x}'_i}{\partial \Theta} \quad (6.9)$$

$$= \frac{\omega_i \hat{x}_i}{\hat{x}^+ + \sum_{i=1}^N \omega_i \cdot \hat{x}_i} \cdot \frac{\partial \hat{x}_i}{\partial \Theta} \quad (6.10)$$

式 (6.10) 是微分的链式法则的结果，第一项 $\frac{\omega_i \hat{x}_i}{\hat{x}^+ + \sum_{i=1}^N \omega_i \cdot \hat{x}_i}$ 由损失函数的具体形式决定的，第二项 $\frac{\partial \hat{x}_i}{\partial \Theta}$ 由编码器和评分函数的形式决定的。可以看到，重要性权重 ω_i 出现在第一项的分子，作用类似一个调节器，它直接控制了未标注样本 x'_i 对模型参数更新的贡献大小，进而控制了这些样本被推多远。

从正例对齐、负例均匀的任务目标考虑，合意的权重 ω_i 应该满足如下要求：

- 如果未标注样本 x'_i 是一个和锚点同类的伪负例，合意的权重 ω_i 应该取值较小，最好是 0，以防止正样本被推远，使得同类样本距离较近，体现伪负例去偏原则。伪负例可以看作负样本中出现的噪声，较小的 ω_i 可以使得式 (6.10) 的梯度消失，能够防止模型学到噪声样本的模式。
- 如果未标注样本 x'_i 是一个和锚点不同类的真负例，合意的权重 ω_i 应该取值较大，以推远硬负样本，使得不同类样本距离较远，体现硬负例挖掘原则。困难负样本可以看作难以拟合的负样本，较大的 ω_i 可以使得式 (6.10) 的梯度幅度增加，激励模型去学习难以区分的样本之间的决策边界。

综合上面的分析，合意的 ω_i 与相应的未标注样本 x'_i 的所属类别相联系：未标注

样本 x'_i 是真负例的概率越低，权重 ω_i 值应该越小（伪负例去偏要求）；未标注样本 x'_i 是真负例的概率越高，权重 ω_i 应该越大（硬负例挖掘要求）。下面章节将介绍如何计算一个合意的重要性权重，满足上述要求。

6.3.2 重要性权重计算

6.3.2.1 相似度分数分布

根据重要性采样的规定步骤，需要目标抽样群体的硬负例的分布，和实际抽样群体的未标注样本的分布，那么重要权重为目标抽样群体和实际抽样群体分布的密度比。为了得到这个权重，首先聚焦于未标注样本与锚点的相似度分数 \hat{x} ，它更一般的形式化表示方式为 $\exp(\text{sim}(f(x), f(x'))/t)$ ，其中 $f(\cdot)$ 是基于深度神经网络的编码器编码的表示向量， $\text{sim}(f(x), f(x'))$ 表示锚点嵌入和未标记样本嵌入的相似度，如内积余弦相似度等。 t 为温度系数。

视未标注样本 x' 为随机变量，则相似度分数 $\hat{x} = \exp(\text{sim}(f(x), f(x'))/t)$ 的分布的影响因素众多：与锚点 x 的选择有关，与神经网络 $f(\cdot)$ 的结构有关，也与相似度函数 $\text{sim}(\cdot, \cdot)$ 的具体形式有关。第3章的引理 3.1 讨论了，在取似度函数 $\text{sim}(\cdot, \cdot)$ 为向量内积时，即使是简单的两个高斯随机向量的相似度，其分布表达式也是非常复杂的，经过非线性函数 $\exp(\cdot/t)$ 映射以后，想获取其密度表达式更加困难。况且，假设基于深度神经网络学到的表示向量 $f(x)$ 服从高斯分布也及其牵强。因此，第3章的方法，以及现有的研究^[132] 通过高斯混合模型^[133]、Beta 混合模型^[132] 等参数化方法来拟合未标注样本的相似度分数 \hat{x} 分布，引入了过强的假设，不具一般性。此外，用于密度估计的学习算法是昂贵的，因为混合系数是隐藏变量，只能通过 EM 算法^[113] 的迭代数值计算来获得，且算法对初始值非常敏感。

本节提出了一种方法，无需显式地估计相似度分数分布，即可实现对密度比的解析计算。对于固定的锚点，假设该锚点与其他未标注样本的相似度分数 \hat{x} 独立同分布于某个锚点特定的未知分布 ϕ ，即

$$\hat{x} \sim \phi(\hat{x}) \quad (6.11)$$

相应的累积分布函数 $\Phi(\hat{x}) = \int_{-\infty}^{\hat{x}} \phi(t) dt$ 。注意，分布 ϕ 是特定于锚点的，不同锚点对应的相似度分数分布不必相同。由于不对分布 ϕ 做出任何假设，因此本方法属于

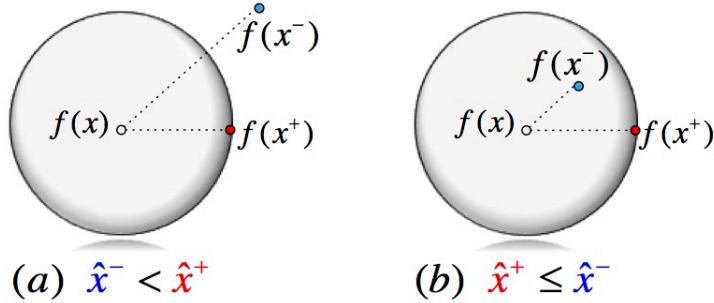


图 6.3 正负样本对相对位置示意图

非参数方法。根据上述抽象表达式，可以写出它对应的次序统计量分布为：

$$\phi_{(1)}(\hat{x}) = 2\phi(\hat{x})[1 - \Phi(\hat{x})] \quad (6.12)$$

$$\phi_{(2)}(\hat{x}) = 2\phi(\hat{x})\Phi(\hat{x}) \quad (6.13)$$

6.3.2.2 目标采样群体

对于固定的锚点 x ，若随机样本是伪负例 $x' \in F_N$ ，则记该未标注样本与锚点的相似度分数 $\hat{x} \in F_N$ ，表明这是一个来自伪负例的相似度分数，所有的伪负例相似度分数总体对应的类条件概率密度记为 ϕ_{F_N} 。类似地，若随机样本是真负例 $x' \in T_N$ ，则记该未标注样本与锚点的相似度分数 $\hat{x} \in T_N$ ，表明这是一个来自真负例的相似度预测分数，所有的真负例相似度分数总体的类条件概率密度记为 ϕ_{T_N} 。

接下来，可以通过正样本和负样本的表示在超球面上的相对位置，解析真负例的类条件概率密度 ϕ_{T_N} 表达式。考虑一个锚点，正例和真负例构成的 (x, x^+, x^-) 三元组，存在一个以 $f(x)$ 为中心、半径为 d^+ 的闭球 $\mathfrak{B}[f(x), d^+] = \{f(\cdot) | d(f(x), f(\cdot)) \leq d^+\}$ ，其中 $d^+ = \|f(x) - f(x^+)\|$ 是锚点嵌入 $f(x)$ 和正样本嵌入 $f(x^+)$ 之间的距离。那么，真负例的相对位置有且只有两种可能情况： $f(x^-) \in \mathfrak{B}[f(x), d^+]$ 或 $f(x^-) \notin \mathfrak{B}[f(x), d^+]$ 。图 6.3 描述了这两种可能的情况：图 6.3(a) 对应于正例距离更近，即 $d^+ < d^-$ ；图 6.3(b) 对应于负例距离更近，即 $d^- \leq d^+$ ，其中 $d^- = \|f(x) - f(x^-)\|$ 是欧式距离度量。对于情况 (a)，必有 $\hat{x}^- < \hat{x}^+$ ，对于情况 (b)，必有 $\hat{x}^+ \leq \hat{x}^-$ 。这是因为所有嵌入 $f(\cdot)$ 都在半径为 $1/t$ 的超球面上，那么欧氏距离与相似度分数一一对应，存在如下关系： $d^\pm = \sqrt{2/t^2 - 2f(x)^\top f(x^\pm)}$ 。

对于关心的真负例相似度分数 \hat{x}^- ，在图 6.3 中用蓝色进行了标记，在情况 (a) 下，

它是次序统计量 $X_{(1)}$ 的一个取值，在情况 (b) 下是次序统计量 $X_{(2)}$ 的一个取值。记情况 (a) 发生的概率为 α ，情况 (b) 发生的概率为 $1 - \alpha$ 。那么，真负例的相似度分数观测值的生成过程可以描述如下：以概率 α 选择情况 (a)，然后从次序统计量分布 $\phi_{(1)}$ 生成观测值 \hat{x} ；或者以概率 $1 - \alpha$ 选择情况 (b)，然后从次序统计量分布 $\phi_{(2)}$ 生成观测值 \hat{x} 。也就是说，真负例的类条件概率密度是 ϕ_{TN} 是成分 $\phi_{(1)}$ 和成分 $\phi_{(2)}$ 以系数 α 进行混合构成：

$$\phi_{TN}(\hat{x}) = \alpha\phi_{(1)}(\hat{x}) + (1 - \alpha)\phi_{(2)}(\hat{x}) \quad (6.14)$$

类似地，伪负例相似度分数的类条件概率密度 ϕ_{FN} 是成分 $\phi_{(2)}$ 和成分 $\phi_{(1)}$ 以系数 α 进行混合构成：

$$\phi_{FN}(\hat{x}) = \alpha\phi_{(2)}(\hat{x}) + (1 - \alpha)\phi_{(1)}(\hat{x}) \quad (6.15)$$

需要说明的是，次序统计量 $X_{(1)}$ 的定义是满足 $X_{(1)} \leq X_{(2)}$ 。在情况 (a) 下 $\hat{x}^- < \hat{x}^+$ ，将真负例的相似度分数 \hat{x}^- 视作次序统计量 $X_{(1)}$ 的一个取值，忽略了 $\hat{x}^- = \hat{x}^+$ 的特殊情况。对于这种特殊情况 \hat{x}^- 的概率测度为 0，因为 ϕ 是由 Lebesgue 测度控制的连续密度函数。直观地理解，对于任意连续概率密度函数如高斯分布，随机变量取任意离散点的概率为 0。下面的引理证明了，类条件概率密度 ϕ_{TN} 和 ϕ_{FN} 满足概率密度函数的非负性和归一性要求。

引理 6.1 (类条件概率密度): 若相似度分数的分布 $\phi(\hat{x})$ 是连续的概率密度函数，满足正定性 $\phi(\hat{x}) \geq 0$ 和归一性 $\int_{-\infty}^{+\infty} \phi(\hat{x})d\hat{x} = 1$ ，那么

- (1) $\phi_{TN}(\hat{x})$ 是概率密度函数，满足 $\phi_{TN}(\hat{x}) \geq 0$ ，且 $\int_{-\infty}^{+\infty} \phi_{TN}(\hat{x})d\hat{x} = 1$ 。
- (2) $\phi_{FN}(\hat{x})$ 是概率密度函数，满足 $\phi_{FN}(\hat{x}) \geq 0$ ，且 $\int_{-\infty}^{+\infty} \phi_{FN}(\hat{x})d\hat{x} = 1$ 。

证明：由于 $\phi(\hat{x}) \geq 0$ 且 $\int_{-\infty}^{+\infty} \phi(\hat{x})d\hat{x} = 1$ ，因此

$$\begin{aligned} \phi_{TN}(\hat{x}) &= \alpha\phi_{(1)}(\hat{x}) + (1 - \alpha)\phi_{(2)}(\hat{x}) \\ &= 2\alpha\phi(\hat{x})[1 - \Phi(\hat{x})] + 2(1 - \alpha)\phi(\hat{x})\Phi(\hat{x}) \\ &\geq 0, \end{aligned} \quad (6.16)$$

其中 $\Phi(\hat{x}) \in [0, 1]$ 。

$$\int_{-\infty}^{+\infty} \phi_{TN}(\hat{x})d\hat{x} = \int_{-\infty}^{+\infty} 2\alpha\phi(\hat{x})[1 - \Phi(\hat{x})] + 2(1 - \alpha)\phi(\hat{x})\Phi(\hat{x})d\hat{x}$$

$$\begin{aligned}
 &= 2\alpha \int_{-\infty}^{+\infty} \phi(\hat{x})[1 - \Phi(\hat{x})]d\hat{x} + 2(1 - \alpha) \int_{-\infty}^{+\infty} \phi(\hat{x})\Phi(\hat{x})d\hat{x} \\
 &= 2\alpha \int_{-\infty}^{+\infty} [1 - \Phi(\hat{x})]d\Phi(\hat{x}) + 2(1 - \alpha) \int_{-\infty}^{+\infty} \Phi(\hat{x})d\Phi(\hat{x}) \\
 &= 2\alpha \int_0^1 (1 - \mu)d\mu + 2(1 - \alpha) \int_0^1 \mu d\mu \tag{6.17} \\
 &= [\alpha(2\mu - \mu^2) + (1 - \alpha)\mu^2] \Big|_0^1 \\
 &= 1,
 \end{aligned}$$

其中，公式 6.17 是对 $\Phi(\hat{x})$ 和 μ 进行换元积分。同样地，

$$\begin{aligned}
 \phi_{FN}(\hat{x}) &= \alpha\phi_{(2)}(\hat{x}) + (1 - \alpha)\phi_{(1)}(\hat{x}) \\
 &= 2\alpha\phi(\hat{x})\Phi(\hat{x}) + 2(1 - \alpha)\phi(\hat{x})[1 - \Phi(\hat{x})] \\
 &\geq 0,
 \end{aligned} \tag{6.19}$$

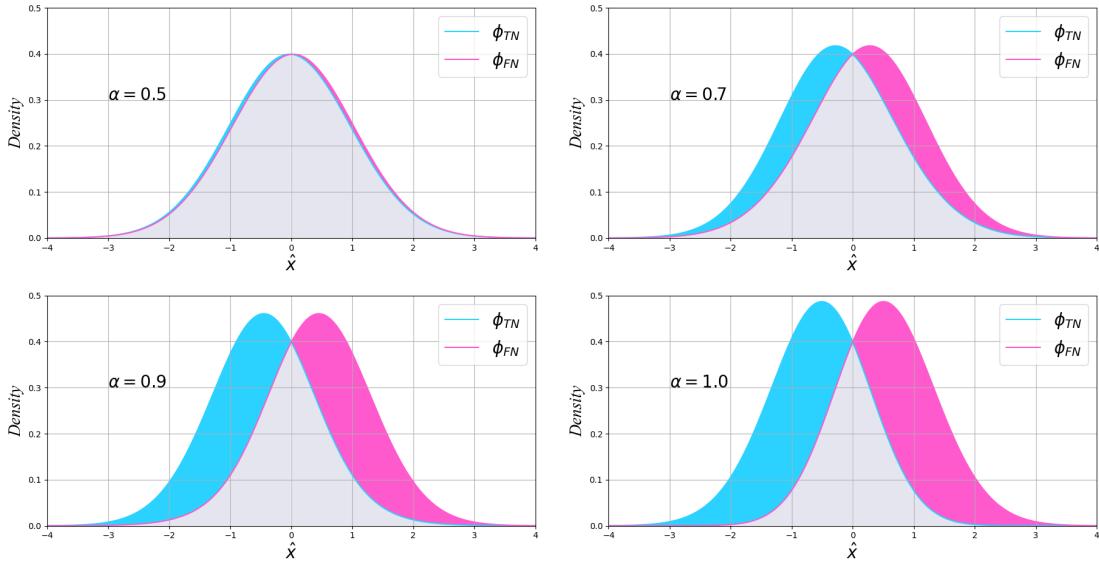
且

$$\begin{aligned}
 \int_{-\infty}^{+\infty} \phi_{FN}(\hat{x})d\hat{x} &= 2\alpha \int_{-\infty}^{+\infty} \phi(\hat{x})\Phi(\hat{x})d\hat{x} + 2(1 - \alpha) \int_{-\infty}^{+\infty} \phi(\hat{x})[1 - \Phi(\hat{x})]d\hat{x} \\
 &= 2\alpha \int_{-\infty}^{+\infty} \Phi(\hat{x})d\Phi(\hat{x}) + 2(1 - \alpha) \int_{-\infty}^{+\infty} [1 - \Phi(\hat{x})]d\Phi(\hat{x}) \\
 &= 2\alpha \int_0^1 \mu d\mu + 2(1 - \alpha) \int_0^1 (1 - \mu)d\mu \\
 &= [\alpha\mu^2 + (1 - \alpha)(2\mu - \mu^2)] \Big|_0^1 \\
 &= 1.
 \end{aligned} \tag{6.20}$$

证毕。 \square

通过查看图 6.3，有助于进一步理解混合系数 α 的含义： α 描述了情况 (a) 发生的概率，即编码器将伪负例（正例）编码到比负例更近的距离的概率，也即正例评分高于负例评分的概率。对于一个随机盲猜的最差编码器 f ， $\alpha = 0.5$ ；而对于一个完美的编码器 $\alpha = 1$ 。因此，合理的 α 值应在 $[0.5, 1]$ 范围内。实际上， α 的含义就是当前训练时点的编码器的 AUC，它对应于数据集 \mathcal{D} 中所有锚点 x 宏平均的 AUC 经验估计：

$$\alpha = \int_{x \in \mathcal{X}} \int_0^{+\infty} \int_0^{+\infty} \mathbb{I}(\hat{x}^+ \geq \hat{x}^-) p(\hat{x}^+, \hat{x}^-) p(x) d\hat{x}^+ d\hat{x}^- dx$$


 图 6.4 不同 α 设定下类条件概率密度示意图

$$\begin{aligned} &\simeq \frac{1}{|\mathcal{D}|} \frac{1}{|\mathcal{D}^+||\mathcal{D}^-|} \sum_{\mathcal{D}^+} \sum_{\mathcal{D}^-} \mathbb{I}(\hat{x}^+ \geq \hat{x}^-) \\ &= \frac{1}{|\mathcal{D}|} AUC \end{aligned} \quad (6.21)$$

因此 α 的值可以通过少部分标记的样本进行经验估计，或者作为超参数。通过在公式 (6.12) 和公式 (6.13) 中将相似度分数分布 $\phi(\hat{x})$ 设定为 $\mathcal{N}(0, 1)$ ，可以作为一个实例，帮助了解 α 如何影响公式 (6.14) 和公式 (6.15) 中的类条件概率密度 $\phi_{TN}(\hat{x})$ 和 $\phi_{FN}(\hat{x})$ ，如图 6.4 所示。较大的 α 值导致 $\phi_{TN}(\hat{x})$ 与 $\phi_{FN}(\hat{x})$ 之间的区分度更高，因为更好的编码器会将具有不同类标签的样本编码得更正交^[97]。需要说明的是，将布 $\phi(\hat{x})$ 设定为 $\mathcal{N}(0, 1)$ 只是为了解释说明，后续计算不涉及任何对 $\phi(\hat{x})$ 的具体形态的假设。

真负例分数的类条件概率密度 $\phi_{TN}(\hat{x})$ ，就是理想的目标抽样群体，描述了来自真负样本群体的相似度分数分布，来自这部分的样本计算得到的损失函数，即为有监督损失。需要关注的是 6.2 中 x'_1 （狼）这个困难负样本示例。它是与锚点类别不同的负样本，但是由于和锚点很相似，难以区分，这种情况对应于图 6.3(b) 中所描述的情况，即真负例距离锚点更近，属于被错分的样本。这类硬负例的相似度分数密度对应于式(6.14)中的第二项，即 $(1 - \alpha)\phi_{(2)}(\hat{x})$ ，称为困难负样本成分；另外一类容易负样本如 6.2 中 x'_5 （猫），它容易区分，对应于图 6.3(a) 中所描述的情况，这类容易负样本的相似度分数密度对应于式(6.14)中的第一项，称为容易负样本成分。

华 中 科 技 大 学 博 士 学 位 论 文

困难负样本由于与锚点非常相似，难以区分，总是被编码到锚点比较近的距离，不利于下游分类任务。为了加大对真负例总体 $\phi_{TN}(\hat{x})$ 中困难负样本成分 $(1 - \alpha)\phi_{(2)}(\hat{x})$ 的惩罚力度，引入一个参数 $\beta \in [0.5, 1]$ ，对真负例的群体 $\phi_{TN}(\hat{x})$ 进行细分，从困难负样本成分中抽样 β 的比例，从容易负样本成分中抽样 $1 - \beta$ 的比例，构成新的抽样目标：

$$\phi_{THN}(\hat{x}) = (1 - \beta)\alpha\phi_{(1)}(\hat{x}) + \beta(1 - \alpha)\phi_{(2)}(\hat{x}) \quad (6.22)$$

参数 $1 - \beta$ 控制着易负样本成分 $\alpha\phi_{(1)}(\hat{x})$ 的比例，而 β 控制着被分类器错误分类的难负样本成分 $(1 - \alpha)\phi_{(2)}(\hat{x})$ 的比例。 $\beta = 1$ 时，新的目标抽样群体 $\phi_{THN}(\hat{x})$ 只有困难负样本成分； $\beta = 0.5$ 时，容易负样本成分和困难负样本的比例与原始抽样目标是一致的； $\beta = 0$ 时，新的目标抽样群体 $\phi_{THN}(\hat{x})$ 只有容易负样本成分；因此，公式 (6.22) 可以解释为具有困难级别 (hardness level) 为 β 的困难负样本分布，较大的 β 值（接近 1）导致中包含更高比例的被分类器错误分类的困难负样本成分。

6.3.2.3 实际采样群体

实际采样的样本来自未标注样本，通过正样本和负样本的类条件密度，可以应用全概率公式将边际概率与条件概率联系起来，从而获得未标记数据的密度 ϕ_{UN}

$$\phi_{UN} = \tau^- \phi_{TN} + \tau^+ \phi_{FN} \quad (6.23)$$

未标注样本的概率密度决定性因素是伪负例的类先验概率 τ 。一个伪负例占比为 50% 的数据集，和伪负例占比为 5% 的数据集，相似度的分数分布差别是很大的。

为了避免混淆，下表梳理了本章出现的密度的符号及含义。

表 6.1 概率密度一览表

符号	含义	描述	是否包含参数化假设
$\phi(\hat{x})$	特定于锚点的相似度分数分布	由锚点 x ，相似度函数，以及编码器结构 f 决定	否
$\phi_{(1)}(\hat{x})$	次序统计量 $X_{(1)}$ 的分布	由 $\phi(\hat{x})$ 决定	否
$\phi_{(2)}(\hat{x})$	次序统计量 $X_{(2)}$ 的分布	由 $\phi(\hat{x})$ 决定	否
$\phi_{TN}(\hat{x})$	真负例相似度分数的类条件概率密度	目标抽样群体	否
$\phi_{FN}(\hat{x})$	伪负例相似度分数的类条件概率密度	-	否
$\phi_{THN}(\hat{x})$	硬负例的相似度分数分布	通过细分 $\phi_{TN}(\hat{x})$ 得到的 新的目标抽样群体	否
$\phi_{UN}(\hat{x})$	未标记样本的相似度分数分布	实际抽样群体	否

需要特别强调的是相似度分数分布 $\phi(\hat{x})$ 与未标记样本的相似度分数分布 $\phi_{UN}(\hat{x})$ 的区别：前者由锚点 x ，相似度函数，以及编码器结构 f 决定，包含了所有的可能的类先验 τ 可能的取值的分布；而后者未标记样本的相似度分数分布 $\phi_{UN}(\hat{x})$ 由特定的类先验 τ 决定。

6.3.2.4 蒙特卡洛重要性采样

上面的分析只得到了目标抽样群体和实际抽样群体密度函数的抽象表达式，但是足以计算重要性权重 ω_i 。根据经典的蒙特卡洛重要性采样^[134,135]的计算步骤，重要性权重的计算公式为目标抽样群体与实际抽样群体的密度比

$$\omega_i(\hat{x}_i; \alpha, \beta) = \frac{\phi_{THN}(\hat{x}; \alpha, \beta)/Z}{\phi_{UN}(\hat{x})} \quad (6.24)$$

$$= \frac{1}{Z} \frac{(1-\beta)\alpha + (\beta-\alpha)\Phi(\hat{x})}{\alpha\tau^- + (1-\alpha)\tau^+ + (1-2\alpha)\Phi(\hat{x})(\tau^- - \tau^+)} \quad (6.25)$$

其中 Z 为目标抽样分布 ϕ_{THN} 的归一化配分因子，可以通过边际积分准确计算如下：

$$\begin{aligned} Z &= \int_{-\infty}^{\infty} \phi_{THN}(\hat{x}) d\hat{x} \\ &= \int_{-\infty}^{\infty} [(1-\beta)\alpha\phi_{(1)}(\hat{x}) + \beta(1-\alpha)\phi_{(2)}(\hat{x})] d\hat{x} \\ &= (1-\beta)\alpha + \beta(1-\alpha) \end{aligned} \quad (6.26)$$

可以看到式(6.25)中的重要性权重 $\omega_i(\hat{x}_i; \alpha, \beta)$ 是累积分布函数 (CDF) $\Phi(\hat{x})$ 的函数，其中密度函数 $\phi(\hat{x})$ 由于式(6.25)的分式形式，可以通过约分被消去，进而无需对密度函数的具体形式做出参数化假设。只需要 $\Phi(\hat{x})$ 的值，就可以计算相应的重要性权重。

下面介绍如何计算累积分布函数 (CDF) $\Phi(\hat{x})$ 。对公式(6.23)等号两侧进行积分，可得：

$$\begin{aligned} \int_{-\infty}^{\hat{x}} \phi_{UN}(t) dt &= \int_{-\infty}^{\hat{x}} \tau^- \phi_{TN}(t) + \tau^+ \phi_{FN}(t) dt \\ &= \int_{-\infty}^{\hat{x}} 2\phi(t)[\alpha\tau^- + (1-\alpha)\tau^+ + (1-2\alpha)\Phi(t)(\tau^- - \tau^+)] dt \end{aligned} \quad (6.27)$$

$$\begin{aligned} &= 2[\alpha\tau^- + (1-\alpha)\tau^+] \int_{-\infty}^{\hat{x}} \phi(t) dt + (1-2\alpha)(\tau^- - \tau^+) \int_{-\infty}^{\hat{x}} 2\phi(t)\Phi(t) dt \\ &= 2[\alpha\tau^- + (1-\alpha)\tau^+] \Phi(\hat{x}) + (1-2\alpha)(\tau^- - \tau^+) \Phi^2(\hat{x}) \end{aligned} \quad (6.28)$$

华 中 科 技 大 学 博 士 学 位 论 文

式(6.27)等号左侧对密度函数积分，正是累积分布函数 $\Phi_{UN}(\hat{x})$ 。记

$$a = (1 - 2\alpha)(\tau^- - \tau^+)$$

$$b = 2[\alpha\tau^- + (1 - \alpha)\tau^+]$$

那么式(6.28)可以简化为一个一元二次方程

$$\Phi_{UN}(\hat{x}) = a\Phi^2(\hat{x}) + b\Phi(\hat{x})$$

根据求根公式有

$$\Phi(\hat{x}) = \frac{-b + \sqrt{b^2 + 4a\Phi_{UN}(\hat{x})}}{2a} \quad (6.29)$$

另外一个解 $\Phi(\hat{x}) = \frac{-b - \sqrt{b^2 + 4a\Phi_{UN}(\hat{x})}}{2a}$ 小于 0 舍去，因为它不在累积分布函数的取值区间内。上式建立了两个累积分布函数之间的关系，只要给出未标注样本的累积分布函数 $\Phi_{UN}(\hat{x})$ ，就可以计算得到所需的 $\Phi(\hat{x})$ ，从而用于计算重要性权重。

最后只剩下如何计算未标注样本的累积分布函数 $\Phi_{UN}(\hat{x})$ 。可以使用如下经验分布函数进行近似：

$$\hat{\Phi}_{UN}(\hat{x}) = \frac{1}{N} \sum_{i=1}^N \mathbb{I}_{|X_i \leq \hat{x}|}, \quad (6.30)$$

根据根据 Glivenko 定理^[100] 所述，经验累积分布函数 $\hat{\Phi}_{UN}(\hat{x})$ 收敛于标准的累积分布函数 (C.D.F) $\Phi_{UN}(\hat{x}) = \int_{-\infty}^{\hat{x}} \phi_{UN}(t) dt$ 。

到目前为止，重要性权重的计算已经介绍完毕。它们的计算只涉及以下内容：(i) $\hat{\Phi}_{UN}(\hat{x}) \in [0, 1]$ ，它代表了贝叶斯视角下的样本信息，可以直接从 N 个未标注样本计算得出。它包含了当前模型对未标注样本是伪负例 $\hat{x} \in F_N$ 的概率解释：对于较大的 \hat{x} 值（与锚点的嵌入距离更接近）， $\hat{\Phi}_{UN}(\hat{x})$ 输出一个更接近于 1 的值，表明 \hat{x} 是一个来自伪负例的相似度观测值。换言之， $\hat{\Phi}_{UN}(\hat{x})$ 反映了当前模型对未标注样本的分类结果。(ii) 类别先验概率 τ 。由样本信息和先验信息计算到的权重 ω_i 自然包含后验信息，因此把经过上述重要性权重加权的自监督对比损失成为贝叶斯自监督对比损失 (Bayesian self-supervised contrastive loss, BCL)。定理 6.1 分析了权重 ω 与样本未真负例的后验概率之间的联系，它们是严格的线性关系。

参数 $\alpha \in [0.5, 1]$ 对应于编码器的宏 AUC 度量，控制着样本信息的置信水平。在训练过程中，可以通过验证数据集进行经验估计。参数 $\beta \in [0.5, 1]$ 对应于真负例群

体中被错误分类的困难负样本成分的比例，是一个根据任务场景所需的硬负例挖掘困难等级 (hardness level) 决定的超参数。在计算对比损失时，还是使用从数据分布中采样的未标记样本 x'_i ，但使用相应的重要性权重 $\omega(\hat{x}, \alpha, \beta)$ 进行加权，以校正目标采样群体和实际采样群体之间的偏差。那么，给定 N 个随机负采样的未标注样本 $\{x'_i\}_{i=1}^N$ ，校正后的损失函数形式为

$$\mathcal{L}_{\text{BCL}} = \mathbb{E}_{\substack{x \sim p_d \\ x^+ \sim p^+ \\ x' \sim p_d}} \left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \sum_{i=1}^N \omega_i \cdot e^{f(x)^T f(x'_i)}} \right] \quad (6.31)$$

6.4 算法实现与时间复杂度分析

6.4.1 算法步骤

重要性权重计算的逻辑有一些复杂，主要是解析目标采样分布和实际采样分布的抽象函数表达式，但计算步骤是很简单的，相对于标准的对比损失函数，只涉及三个额外计算步骤，如下伪代码所示。

Algorithm 5 贝叶斯自监督对比学习算法 (BCL) 伪代码

Input: 数据条目组织形式为 (锚点 x , 正例 x^+ , N 个随机负采样的未标注样本 x'_1, \dots, x'_n) 的训练集 \mathcal{D} , 正例类先验 τ^+ , 去偏超参数 α , 硬负例难度超参数 β 。

Output: 模型参数 $\Theta \in \mathbb{R}^d$

计算正例分数 $\hat{x}^+ = e^{f(x)^T f(x^+)}$;

计算 N 个未标注样本分数 $\hat{x}_i = e^{f(x)^T f(x'_i)}$, $i = 1, 2, \dots, N$;

for $i = 1, 2, \dots, N$ **do**

通过公式(6.30)计算经验分布函数 $\hat{\Phi}_{\text{UN}}(\hat{x}_i)$;

通过公式(6.29)计算分布函数 $\Phi(\hat{x}_i)$;

通过公式(6.25)计算重要性权重 ω_i ;

计算损失函数 $\mathcal{L} = -\log \frac{\hat{x}^+}{\hat{x}^+ + \sum_{i=1}^N \omega_i \cdot \hat{x}_i}$;

根据 \mathcal{L} 相对于 Θ 的梯度更新参数 Θ ;

Result: 模型参数 Θ 。

将数据条目组织为 (锚点 x , 正例 x^+ , N 个随机负采样的未标注样本 x'_1, \dots, x'_n) 的元组结构，可以通过重写 Pytorch 的 `get_item` 方法或 `collate_fn` 方法实现，二者类似。第5章的算法3提供了重写 `collate_fn` 方法的算法伪代码，本章不再赘述。

6.4.2 时间复杂度分析

本章所介绍的方法，是上一章从 $N = 1$ 个负例向多个负例的更一般的对比损失的推广，但基本思想都是估计校正，不涉及显式的负采样，仍然使用 in-batch 内的

随机负样本计算对比损失，因此也不涉及对 mini-batch 以外的额外计算和存储开销。由于要计算重要性权重，从而对对比损失进行校正，从而引入了额外的时间复杂度。下面以标准的 InfoNCE 损失为基准，分析本章所提出的方法的计算的额外开销。

在计算标准的 InfoNCE 损失时，也需要正向计算正例的相似度分数以及未标注样本的相似度分数，然后反向传播按照梯度下降算法更新参数，那么 InfoNCE 也要执行算法5的for循环内的最后两行。因此贝叶斯自监督对比学习的额外计算开销主要是：(1) 通过公式(6.30)计算经验分布函数 $\hat{\Phi}_{\text{UN}}(\hat{x}_i)$ ，时间复杂度为 $\mathcal{O}(N)$ ；(2) 通过公式(6.29)计算分布函数 $\Phi(\hat{x}_i)$ ，时间复杂度为 $\mathcal{O}(1)$ ；(3) 通过公式(6.25)计算重要性权重 ω_i ，时间复杂度为 $\mathcal{O}(1)$ 。计算重要性权重的公式(6.25)尽管看起来很复杂，但是都是标量计算，因此是常数时间复杂度。因此，一个样本的重要性权重 ω_i 额外计算开销为 $\mathcal{O}(N)$ ，通常远低于编码样本 $f(x)$ 的时间复杂度，是可以忽略的。在本章实验部分对比了实际的运行时间，本方法和基准方法 InfoNCE 相比，有着几乎相同的实际运行时间。

6.5 贝叶斯自监督对比损失的理论分析

本章的6.3.1小节分析了，基于对比学习“正例对齐、负例均匀”的任务考量，合理的权重 ω_i 与相应的未标注样本 x'_i 的所属类别相联系：未标注样本 x'_i 是真负例的概率越低，权重 ω_i 值应该越小（伪负例去偏要求）；未标注样本 x'_i 是真负例的概率越高，权重 ω_i 应该越大（硬负例挖掘要求）。那么，本章的 ω_i 值是否满足上述要求？本章计算的 ω_i 与样本是真负例的概率有什么联系？此外，经过对随机样本的重要性加权以后计算得到的贝叶斯自监督对比损失，与完全监督数据计算得到的对比损失有什么联系？

为了回答上面的问题，首先考虑一个如图6.5描述的抽象数学模型：图6.5(a) 表示未标注样本 x'_i 与锚点的相似度观测值 \hat{x} 为随机变量，假设它独立同分布于某个未知分布 ϕ ，图6.5(b) 表示未标注样本观测值中，来自正类的占比为 τ^+ ，相应的观测值记为 \hat{x}^+ ；未标注样本观测值中，来自负类的占比为 $\tau^- = 1 - \tau^+$ ，相应的观测值记为 \hat{x}^- 。图6.5(c) 表示正例的观测值以 α 的概率小于负例的观测值。

图6.5(a) 是一个并不严格的假设，因为并没有对未知分布 ϕ 的具体形态做出规定。对于不同的锚点，锚点与未标记样本的相似度分数观测值的分布 ϕ 可以不同；

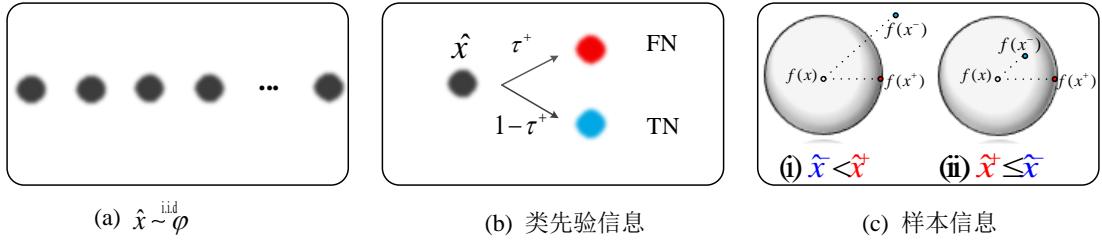


图 6.5 抽象数学模型示意图

图6.5(b) 不是假设，而是已知条件。正例类先验概率 τ^+ 通常是可以获得的，例如在类别均匀的 10 分类数据集中，遇到一个与锚点同类(狗)的同类样本的占比，就是 $1/10$ 。更多面向复杂场景如类别不平衡、PU 数据集的类先验 τ^+ 的估计方法可以参考文献^[126,127]；图6.5(c) 也不是假设，而是已知条件。前面的章节讨论了， α 的含义为当前模型参数下编码器的 AUC，可以通过少部分标注样本进行经验估计，或者设定为超参数。

定理 6.1 (后验概率估计): 设 \hat{x} 独立同分布于某未知分布 ϕ 。其中，正例 \hat{x}^+ 的占比为 τ^+ ，且 $\mathbb{P}(\hat{x}^- < \hat{x}^+) = \alpha$ 。取 $\beta = 0.5$ ，则重要性权重 ω_i 与样本为真负例的后验概率 $P(\text{TN}|\hat{x}_i)$ 存在如下关系

$$\omega_i = \mathbb{P}(\text{TN}|\hat{x}_i)/\tau^-$$

证明： 对于独立同分布于 ϕ 的随机变量 \hat{x} ，其次序统计量 $X_{(1)}$ 和 $X_{(2)}$ 的分布为：

$$\begin{aligned}\phi_{(1)}(\hat{x}) &= 2\phi(\hat{x})[1 - \Phi(\hat{x})] \\ \phi_{(2)}(\hat{x}) &= 2\phi(\hat{x})\Phi(\hat{x})\end{aligned}$$

其中 $\Phi(\hat{x}) = \int_{-\infty}^{\hat{x}} \phi(t)dt$ 是概率密度函数 $\phi(\hat{x})$ 对应的累积分布函数。

由于 $\mathbb{P}(\hat{x}^- < \hat{x}^+) = \alpha$ ，则

$$\mathbb{P}(\hat{x}^- \geq \hat{x}^+) = 1 - \alpha$$

那么 \hat{x} 中的负例类别 \hat{x}^- 的概率密度是次序统计量 $X_{(1)}$ 和 $X_{(2)}$ 的概率密度以系

数 α 进行混合

$$\phi_{TN}(\hat{x}) = \alpha\phi_{(1)}(\hat{x}) + (1 - \alpha)\phi_{(2)}(\hat{x})$$

其中，负例类别 \hat{x}^- 的概率密度，也称类条件概率密度。类似地，正例类别 \hat{x}^+ 的类条件概率密度为

$$\phi_{FN}(\hat{x}) = \alpha\phi_{(2)}(\hat{x}) + (1 - \alpha)\phi_{(1)}(\hat{x})$$

那么，观测到的未标注样本的概率密度为

$$\phi_{UN}(\hat{x}) = \tau^-\phi_{TN}(\hat{x}) + \tau^+\phi_{FN}(\hat{x})$$

其中 $\tau^- = 1 - \tau^+$ 代表负类的占比。以上是本章方法部分已有结论，不再详细论述。那么式(6.24)给出的 $\omega_i(\hat{x}_i; \alpha, \beta = 0.5)$ 可以计算为

$$\begin{aligned} \omega_i &= \frac{\phi_{THN}(\hat{x}; \alpha, \beta = 0.5)/Z}{\phi_{UN}(\hat{x})} \\ &= \frac{(1 - 0.5)\alpha\phi_{(1)}(\hat{x}) + 0.5(1 - \alpha)\phi_{(2)}(\hat{x})}{(1 - 0.5)\alpha + 0.5(1 - \alpha)} \cdot \frac{1}{\phi_{UN}(\hat{x})} \\ &= [\alpha\phi_{(1)}(\hat{x}) + (1 - \alpha)\phi_{(2)}(\hat{x})] \cdot \frac{1}{\phi_{UN}(\hat{x})} \\ &= \phi_{TN}(\hat{x})/\phi_{UN}(\hat{x}) \end{aligned} \quad (6.32)$$

上式结论的直观解释是，如果 $\beta = 0.5$ ，那么 $1 - \beta = \beta$ ，即等比例地从目标抽样群体 $\phi_{TN}(\hat{x})$ 中选取容易负样本成分 $\alpha\phi_{(1)}(\hat{x})$ 和困难负样本成分 $(1 - \alpha)\phi_{(2)}(\hat{x})$ ，那么新的目标抽样群体 $\phi_{THN}(\hat{x}; \alpha, \beta = 0.5)/Z$ 的概率密度与真负例的类条件概率密度 $\phi_{TN}(\hat{x})$ 相等，从而重要性权重为真负例类条件概率密度 $\phi_{TN}(\hat{x})$ 与未标注样本概率密度 $\phi_{UN}(\hat{x})$ 的比值。

对式6.32的结果进行简单的代数变换，有

$$\omega_i = \frac{\phi_{TN}(\hat{x}) \cdot \tau^-}{\phi_{UN}(\hat{x})} \cdot \frac{1}{\tau^-} \quad (6.33)$$

$$= \frac{\phi_{TN}(\hat{x})\tau^-}{\tau^-\phi_{TN}(\hat{x}) + \tau^+\phi_{FN}(\hat{x})} \cdot \frac{1}{\tau^-} \quad (6.34)$$

$$= p(TN|\hat{x}_i) \frac{1}{\tau^-} \quad (6.35)$$

式(6.33)是先乘以负例的类先验 τ^- ，然后再除以负例的类先验。式(6.34)中，第一项分子类条件概率密度 $\phi_{TN}(\hat{x})$ 的含义是真负例类的条件概率密度 $p(\hat{x}|TN)$ ，乘以负例的类先

华 中 科 技 大 学 博 士 学 位 论 文

验 τ^- , 得到联合概率分布 $p(\hat{x}, \text{TN})$ 。同样地, 分母的含义为 $p(\hat{x}, \text{TN}) + p(\hat{x}, \text{FN}) = p(\hat{x})$ 。因此, 式(6.35)正是贝叶斯公式的结果。证毕。 \square

定理6.1的结果表明, 本方法计算得到的重要性权重与样本为真负例的后验概率之间的严格线性关系。一个未标注样本是真负例的后验概率越低, 计算得到的权重 ω_i 越小, 满足伪负例去偏原则; 一个未标注样本是真负例的后验概率越低, 计算得到的权重 ω_i 越大, 满足硬负例挖掘原则。因此, 本方法计算得到的重要性权重本章6.3.1小节所分析的合意的重要性权重的要求。这也是本方法与现有的基于重要性加权的方法 HCL^[98] 的主要区别, HCL 的权重只和未标注样本相似度分数 \hat{x}^β 相联系, 相似度分数越高的样本分配的权重越大。

定理 6.2 (渐进一致估计): 设 \hat{x} 独立同分布于某未知分布 ϕ 。其中, 正例 \hat{x}^+ 的占比为 τ^+ , 且 $\mathbb{P}(\hat{x}^- < \hat{x}^+) = \alpha$ 。取 $\beta = 0.5$, 当负例个数 $N \rightarrow \infty$ 时, 有

$$\mathcal{L}_{\text{BCL}} \rightarrow \mathcal{L}_{\text{SUP}}$$

证明: 主要思路是通过勒贝格支配收敛定理对极限运算和积分运算交换运算次序, 然后使用重要采样的性质进行代数变换。

$$\lim_{N \rightarrow \infty} \mathcal{L}_{\text{BCL}} = \lim_{N \rightarrow \infty} \mathbb{E}_{\substack{x \sim p_d \\ x^+ \sim p^+ \\ x' \sim p_d}} - \log \left[\frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \sum_{i=1}^N \omega_i \cdot e^{f(x)^T f(x'_i)}} \right] \quad (6.36)$$

$$= \mathbb{E}_{x \sim p_d} \lim_{N \rightarrow \infty} - \log \left[\frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \sum_{i=1}^N \omega_i \cdot e^{f(x)^T f(x'_i)}} \right] \quad (6.37)$$

$$= \mathbb{E}_{x \sim p_d} \lim_{N \rightarrow \infty} \left[- \log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + N \cdot \mathbb{E}_{\hat{x} \sim \phi_{\text{UN}}} \omega \hat{x}} \right] \quad (6.38)$$

其中, 式(6.36)是先对损失函数求期望 (即积分运算), 然后求极限运算。根据勒贝格控制收敛定理, 对于一个有界的可测函数序列 f_n , 有 $\lim_{n \rightarrow \infty} \int_{\Omega} f_n = \int_{\Omega} \lim_{n \rightarrow \infty} f_n$ 。应用该结论, 可以交换期望算子和极限算子的运算次序, 得到式(6.37)。

式(6.37)中, 极限运算只作用在分母的第二项中, 因为只有分母的第二项包含 N 。由于

$$\begin{aligned} \lim_{N \rightarrow \infty} \sum_{i=1}^N \omega_i \cdot e^{f(x)^T f(x'_i)} &= \lim_{N \rightarrow \infty} \sum_{i=1}^N \omega_i \cdot \hat{x}_i \\ &= N \mathbb{E}_{\hat{x} \sim \phi_{\text{UN}}} \omega \cdot \hat{x} \end{aligned}$$

于是得到式(6.38)。应用式(6.32)的结论，有

$$\omega(\hat{x}; \alpha, \beta) = \frac{\phi_{\text{TN}}(\hat{x})}{\phi_{\text{UN}}(\hat{x})}$$

将上式带入式(6.38)分母中的第二项可得

$$N\mathbb{E}_{\hat{x} \sim \phi_{\text{UN}}} \omega \hat{x} = N \int \omega \hat{x} \phi_{\text{UN}}(\hat{x}) d\hat{x} \quad (6.39)$$

$$= N \int \hat{x} \frac{\phi_{\text{TN}}(\hat{x})}{\phi_{\text{UN}}(\hat{x})} \phi_{\text{UN}}(\hat{x}) d\hat{x} \quad (6.40)$$

$$= N \int \hat{x} \phi_{\text{TN}}(\hat{x}) d\hat{x} \\ = N\mathbb{E}_{x^- \sim p^-} e^{f(x)^T f(x^-)} \quad (6.41)$$

$$= \lim_{N \rightarrow \infty} \sum_{i=1}^N e^{f(x)^T f(x_i^-)} \quad (6.42)$$

式(6.39)是期望的定义，式(6.40)是带入重要性权重的结果，式(6.41)也是期望的定义。这正是重要性采样的基本性质：使用重要性权重加权来自 ϕ_{UN} 总体中的样本，等于目标总体 ϕ_{TN} 的期望。把式(6.42)的结果代回式(6.38)中可得

$$\begin{aligned} \lim_{N \rightarrow \infty} \mathcal{L}_{\text{BCL}} &= \mathbb{E}_{\substack{x \sim p_d \\ x^+ \sim p^+ \\ x^- \sim p^-}} \lim_{N \rightarrow \infty} \left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \sum_{i=1}^N e^{f(x)^T f(x_i^-)}} \right] \\ &= \lim_{N \rightarrow \infty} \mathbb{E}_{\substack{x \sim p_d \\ x^+ \sim p^+ \\ x^- \sim p^-}} \left[-\log \frac{e^{f(x)^T f(x^+)}}{e^{f(x)^T f(x^+)} + \sum_{i=1}^N e^{f(x)^T f(x_i^-)}} \right] \quad (6.43) \end{aligned}$$

$$= \lim_{N \rightarrow \infty} \mathcal{L}_{\text{SUP}} \quad (6.44)$$

式(6.43)是再次运用勒贝格支配收敛定理交换积分和极限的运算顺序。证毕。 \square

上述证明过程体现了 BCL 方法的核心思想：通过重要性加权未标注样本（实际抽样总体）的相似度分数，以近似真负样本（目标抽样总体）的相似度分数的期望值，从而间接地实现对有监督损失的近似。这与第5章通过概率进行近似不同，主要是因为较大的 N 值导致真实的标签有 2^N 可能，难以枚举。但与第5章的落脚点是一致的，都是获取有监督损失的一致估计。

获得一个与有监督损失一致的估计量，是提升弱监督学习或者无监督学习泛化性能的重要途径。损失函数是模型参数 Θ 与训练样本 x 的函数 $\mathcal{L}(x, \Theta)$ ，如果给定样本 x ，无监督损失 $\mathcal{L}(x, \Theta)$ 与有监督损失 $\mathcal{L}_{\text{SUP}}(x, \Theta)$ 对于所有可能的模型参数 Θ 几乎处处 (almost everywhere) 相等，那么有理由相信，给定相同的训练样本，无监督损失

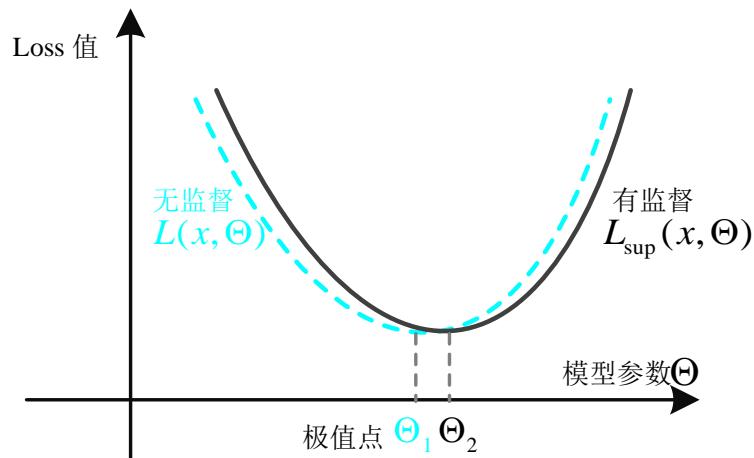


图 6.6 估计校正的直观阐释示意图

$\mathcal{L}(x, \Theta)$ 与有监督损失 $\mathcal{L}_{\text{Sup}}(x, \Theta)$ 有相同的极值点 Θ , 从而在无监督的情况下学到接近于有监督的情况下学到的模型参数, 从而取得与有监督学习类似的泛化性能。这是第5章与本章通过估计校正获取一个与有监督损失一致的估计量的意义所在。

图6.6提供了一个阐释性的例子, 固定训练样本 x , 损失值可以看作模型参数 Θ 的函数。如果, 对于所有的模型参数 Θ 的可能取值, 无监督损失函数和有监督损失函数非常接近, 那么它们的极值点 Θ_1 和 Θ_2 也非常接近, 即在无监督数据集中学到接近于有监督数据的模型参数。但需要注意的是, 无监督损失和有监督损失的一致性往往以样本数量 $N \rightarrow \infty$ 为前提条件, 这是一个理想的情况, 在现实中难以实现, 有限的样本数量导致估计误差的产生。本章中, 在有限样本时, \mathcal{L}_{BCL} 的估计误差具体体现在三个方面: (1) 经验分布函数 $\hat{\Phi}_{\text{UN}}(\hat{x})$ 的估计误差, 它是未知的分布函数 $\Phi_{\text{UN}}(\hat{x})$ 的估计; (2) α 的估计误差, 它是编码器的宏 AUC 估计; (3) 重要性采样的估计误差, 它是真负样本的相似度分数期望值的估计。由于涉及因素众多, 本章不讨论 \mathcal{L}_{BCL} 估计误差的解析表达式。但是, 可以明确的是, 较大的 N 值可以使得上述三个误差都减小, 进而 \mathcal{L}_{BCL} 的估计误差会确定性地减小。在数值实验部分也印证了这个结论。此外, 较大地 N 值会推高互信息地下界。因此, 在实践中, 在 GPU 显存可承受的范围内, 应该选择一个尽可能大的 N 值。

6.6 实验评估

由于 InfoNCE 损失在诸多领域的广泛使用，本章的实验考虑仿真数值实验以及真实数据集实验，在真实数据集上的实验包括个性化推荐任务以及图像分类任务。

6.6.1 数值实验

6.6.1.1 实验设置

前面提到，BCL 是通过重要性加权未标注样本（实际抽样总体）的相似度分数 $\sum_{i=1}^N \omega_i \cdot \hat{x}_i$ ，以近似真负样本（目标抽样总体）的相似度分数的期望值，从而间接地实现对有监督损失的近似。那么，对真负样本（目标抽样总体）的相似度分数的期望值估计质量的好坏，决定了是否能够获取与监督损失一致的估计量。在数值实验部分，探讨对真负样本（目标抽样总体）的相似度分数的期望值估计质量的好坏，它决定了与监督损失的一致性，也体现了权重 ω_i 估计质量的好坏。

前面获得了类条件概率密度（目标抽样群体），以及未标注样本的概率密度（实际抽样群体）的抽象表达式。只要给出任意的相似度分数分布 ϕ ，就可以获得类条件概率密度及未标注样本的概率密度的具体表达式。从真负例的类条件概率密度中，可以生成真负例的样本，从而得到期望值；类似地，从伪负例类条件概率密度中，可以生成伪负例样本。二者共同构成了未标注本。按照 BCL 的计算步骤得到估计值 $\sum_{i=1}^N \omega_i \cdot \hat{x}_i$ 。对比期望值和估计值，可以评估估计质量的好坏。下面介绍样本生成过程。

6.6.1.2 样本生成过程

样本生成过程概述：为了生成样本，首先随机初始化一个相似度分数分布 ϕ ，如均匀分布，高斯分布等。然后就可以生成对应的样本，见图 6.7 所示，共分为三步：

(1) 以正类先验概率 τ^+ 选择伪负例的标签，表明一个观测值以 τ^+ 概率来自伪负例 FN 总体；以负类先验概率 $\tau^- = 1 - \tau^+$ 选择真负例的标签，表明一个观测值以 τ^- 概率来自真负例 TN。

(2) 根据所属类别，从类条件概率密度 ϕ_{FN} （或 ϕ_{TN} ）生成随机样本。需要说明的是，即使将 ϕ 设置为简单分布，相应的类条件密度 ϕ_{FN} （或 ϕ_{TN} ）也不再是简单分布。可以使用接受-拒绝采样^[136] 技术，只要给出概率密度表达式，就可以生成服从相应分布的样本。具体的技术细节在后面介绍。

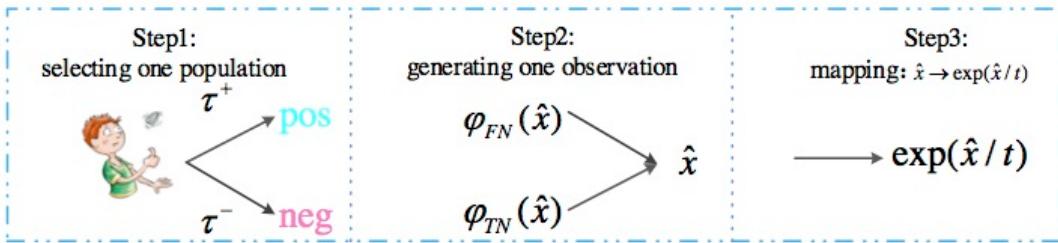


图 6.7 未标注样本相似度分数生成过程示意图

(3) 将生成的样本 \hat{x} 映射为 $\exp(\hat{x}/t)$, 并混合两个类, 作为最终的未标注样本相似度分数观测值。需要说明的是, 前面的方法部分没有讨论将观测值 $\hat{x} \rightarrow \exp(\hat{x}/t)$ 的影响, 这是因为这是一个严格单调的变换, 经过函数映射以后, 经验分布函数保持不变: $\Phi_n(\hat{x}) = \Phi_n(\exp(\hat{x}/t))$, 因此完全不影响 BCL 的估计。

重复图 6.7 描述的过程 N 次, 以生成一个锚点的 N 个观测值, 并对 M 个锚点重复此过程, 就可以得到 M 个锚点所对应的 M 个观测值序列。完整的样本生成过程描述如算法 6 所示。

相似度分数分布 ϕ 可以是任意的, 也称为建议分布 (proposal distribution)。对于投影到半径为 $1/t$ 的超球面的表示, 相似度观测值的值域为 $[-1/t^2, 1/t^2]$ 。为了方便地控制观测值在其理论最小和最大区间 $[-1/t^2, 1/t^2]$ 内, 并且先前的研究表明, 对比损失在渐近情况下优化了负样本的均匀性^[32], 将 ϕ 设置为 $U(a, b)$ 以进行数值实验, 其中 $-1/t^2 \leq a \leq b \leq 1/t^2$ 是针对每个锚点随机选择的。具体而言, 原始区间设置为 $a = -0.5$, $b = 0.5$, 并使用 $\gamma \in [0, 1]$ 来控制从原始区间的变异程度。 $\gamma = 1$ 时, 不同锚点之间参数 a, b 参数变异程度最大; $\gamma = 0$ 时, 不同的参数 a, b 参数变异程度最小, 都为 $a = -0.5$, $b = 0.5$ 。

接受-拒绝采样实现: 生成样本的第二步 (图 6.7), 涉及从从类条件密度 $\phi_{TN}(\hat{x})$ 生成相应的样本 \hat{x} , 使得 $\hat{x} \sim \phi_{TN}(\hat{x})$ 。由于类条件概率密度的表达式通常很复杂, 不再是简单的均匀分布或者高斯分布, 难以直接采样。本节采用标准的接受-拒绝采样方法实现, 其基本思想是从建议分布 ϕ 中生成样本 \hat{x} , 并以接受概率 p_{TN} 接受该样本。为了计算接受概率 p_{TN} , 首先将 $\phi_{TN}(\hat{x})$ 写为 ϕ 的函数

$$\phi_{TN}(\hat{x}) = \alpha\phi_{(1)}(\hat{x}) + (1 - \alpha)\phi_{(2)}(\hat{x})$$

Algorithm 6 未标注样本相似度分数观测值生成过程

Input: 参数 α , 温度系数 t , 未标注样本数量 N , 锚点个数 M

Output: 未标注样本观测值 \hat{x} 及其对应的标签

```
for 锚点  $m = 1, 2, \dots, M$  do
    \\ 选定一个任意参数的相似度分数分布  $\phi$ 
    随机选择  $[a, b] \subseteq [-1/t^2, 1/t^2]$ ;
    设定  $\phi \sim U(a, b)$ ;
    for 负例  $j = 1, 2, \dots, N$  do
        \\ step1: 选择一个总体
         $p = \text{random.uniform}(0, 1)$ ;
        if  $p \leq \tau^+$  then
            \\ step2: 从伪负例类条件分布  $\phi_{FN}$  中生成样本
             $\hat{x}_j = \text{AccRejetSamplingFN}(\phi_{FN})$ ;
            label = False;
        else
            \\ step2: 从真负例类条件分布  $\phi_{TN}$  中生成样本
             $\hat{x}_j = \text{AccRejetSamplingTN}(\phi_{TN})$ ;
            label = True;
            \\ step3: 函数映射
             $\hat{x}_j = \exp(\hat{x}_j/t)$ ;
        收集  $\hat{x}_j$  数值及标签;
```

Result: 未标注样本观测值 \hat{x} 及其对应的标签

华中科技大学博士学位论文

$$= [2\alpha + (2 - 4\alpha)\Phi(\hat{x})]\phi(\hat{x}),$$

其中 $\Phi(\hat{x})$ 是 $\phi(\hat{x})$ 对应的累积分布函数。接下来，寻找一个最小值 c 满足如下不等式

$$c \cdot \phi(\hat{x}) \geq \phi_{TN}(\hat{x}),$$

即

$$c \cdot \phi(\hat{x}) \geq [2\alpha + (2 - 4\alpha)\Phi(\hat{x})]\phi(\hat{x}).$$

因此

$$\begin{aligned} c &= \min[2\alpha + (2 - 4\alpha)\Phi(\hat{x})], \Phi(\hat{x}) \in [0, 1] \\ &= 2\alpha, \end{aligned}$$

最小值 c 在 $\Phi(\hat{x}) = 0$ 时取得，由于 $2 - 4\alpha \leq 0$ 。因此接受概率

$$\begin{aligned} p_{TN} &= \frac{\phi_{TN}(\hat{x})}{c \cdot \phi(\hat{x})} \\ &= [\alpha + (1 - 2\alpha) \cdot \Phi(\hat{x})]/\alpha \end{aligned}$$

从建议分布 $\phi(\hat{x})$ 生成的样本 \hat{x} 以概率 p_{TN} 被接受，得到了服从分布 $\phi_{TN}(\hat{x})$ 的样本，如算法 7 描述。

Algorithm 7 生成服从 ϕ_{TN} 分布的样本的接受拒绝采样算法 AccRejetSamplingTN(ϕ_{TN})
Input: 参数 α , 建议分布 ϕ

Output: 服从分布 ϕ_{TN} 的样本 \hat{x}

从分布 ϕ 中采样一个随机样本 \hat{x}_j ;

$cdf = \int_{-\infty}^{\hat{x}_j} \phi(t)dt;$

$u = \text{random.uniform}(0, 1);$

while $u > [\alpha + (1 - 2\alpha) \cdot cdf]/\alpha$ **do**

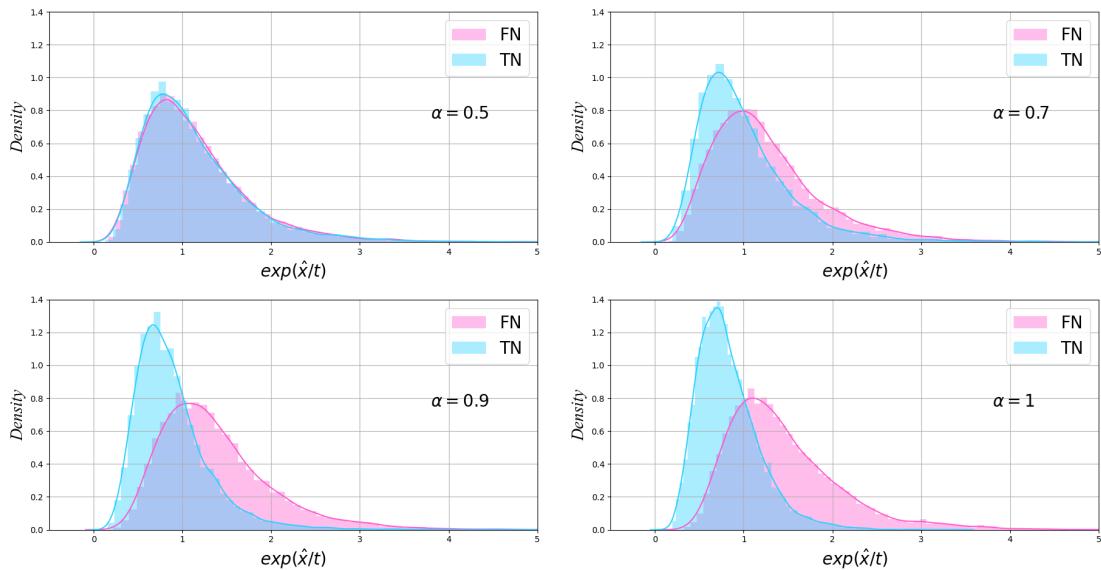
\\ 样本 \hat{x}_j 被拒绝，重新采样

从分布 ϕ 中采样一个随机样本 \hat{x}_j ;

$cdf = \int_{-\infty}^{\hat{x}_j} \phi(t)dt;$

$u = \text{random.uniform}(0, 1);$

Result: 服从分布 ϕ_{TN} 的样本 \hat{x}_j


 图 6.8 不同 α 设置下相似度分数的经验分布

同样地，对于类条件概率密度 ϕ_{FN} ，接受概率的计算步骤类似，为

$$p_{FN} = [1 - \alpha + (2\alpha - 1) \cdot \Phi(\hat{x})]/\alpha$$

从建议分布 $\phi(\hat{x})$ 生成的样本 \hat{x} 以概率 p_{FN} 被接受，得到了服从分布 $\phi_{FN}(\hat{x})$ 的样本，如算法 8 描述。

Algorithm 8 生成服从 ϕ_{FN} 分布的样本的接受拒绝采样算法 AccRejectSamplingFN(ϕ_{FN})

Input: 参数 α , 建议分布 ϕ

Output: 服从分布 ϕ_{FN} 的样本 \hat{x} .

从分布 ϕ 中采样一个随机样本 \hat{x}_j ;

$cdf = \int_{-\infty}^{\hat{x}_j} \phi(t) dt$;

$u = \text{random.uniform}(0, 1)$;

while $u > [1 - \alpha + (2\alpha - 1) \cdot cdf]/\alpha$ **do**

 \\ 样本 \hat{x}_j 被拒绝，重新采样

 从分布 ϕ 中采样一个随机样本 \hat{x}_j ;

$cdf = \int_{-\infty}^{\hat{x}_j} \phi(t) dt$;

$u = \text{random.uniform}(0, 1)$;

Result: 服从分布 ϕ_{FN} 的样本 \hat{x}

根据上述生成过程，图 6.8 绘制了生成的样本 $\exp(\hat{x}/t)$ 的经验分布，其中 ϕ 设定为标注正态分布，与图 6.4 相对应。其中， $t = 2$, $M = 1$ 和 $N = 20000$ 。可以看到，图 6.8 中的经验分布展现出与使用实际数据集报告的分布^[98,132] 相似的结构，表明所提出的对样本 \hat{x} 的生成过程描述的有效性。

6.6.1.3 对比方法

在实践中，通常是使用均值作为期望值的经验估计^[7,8,97]，因此使用 N 个随机样本中真负例相似度分数的均值作为真值的参考，可以通过如下方式计算

$$\theta_{\text{SUP}} = \frac{\sum_{i=1}^N \mathbb{I}(\hat{x}_i) \cdot \hat{x}_i}{\sum_{i=1}^N \mathbb{I}(\hat{x}_i)}, \quad (6.45)$$

其中， $\mathbb{I}(x_i)$ 是指示函数，如果 $\hat{x}_i \in \text{TN}$ ，则 $\mathbb{I}(\hat{x}_i) = 1$ ；否则， $\mathbb{I}(\hat{x}_i) = 0$ ，那么式(6.45)分子的含义是真负例分数和，分母是真负例的个数，因此式(6.45)的含义是真负例分数均值。

对比方法如下：

$$\hat{\theta}_{\text{BIASED}} = \frac{1}{N} \sum_{i=1}^N \hat{x}_i \quad (6.46)$$

$$\hat{\theta}_{\text{DCL}} = \frac{1}{N\tau^-} \left(\sum_{i=1}^N \hat{x}_i - N\tau^+ \cdot \frac{\sum_{j=1}^K \hat{x}_j^+}{K} \right) \quad (6.47)$$

$$\hat{\theta}_{\text{BCL}} = \frac{1}{N} \sum_{i=1}^N \omega_i \cdot \hat{x}_i \quad (6.48)$$

式 (6.46) 是标准的 InfoNCE 损失^[7] 对应的估计量，含义为未标注样本的相似度分数均值。式(6.47)是 DCL^[97] 方法所对应的估计量，其含义在第5章的实验部分介绍了，这里不再赘述。此外 HCL^[98] 方法是从任务需求的角度考量，而非从统计量优良性的角度考量。由于上加权困难样本，过度高估真负例分数的均值，在数值实验中不考虑该对比方法。类似地，式(6.48)中，设定 $\beta = 0.5$ ，不考虑额外的硬负例挖掘任务，只考虑 BCL 方法估计量的优良性。

6.6.1.4 实验结果

对于第 m 个锚点，可以计算得到真负例分数均值的真值，记为 $\theta_{\text{SUP},m}$ ；然后按照对比方法计算一个真负例分数的估计值，记为 $\hat{\theta}_m$ 。进而，对于 M 个锚点，可以计

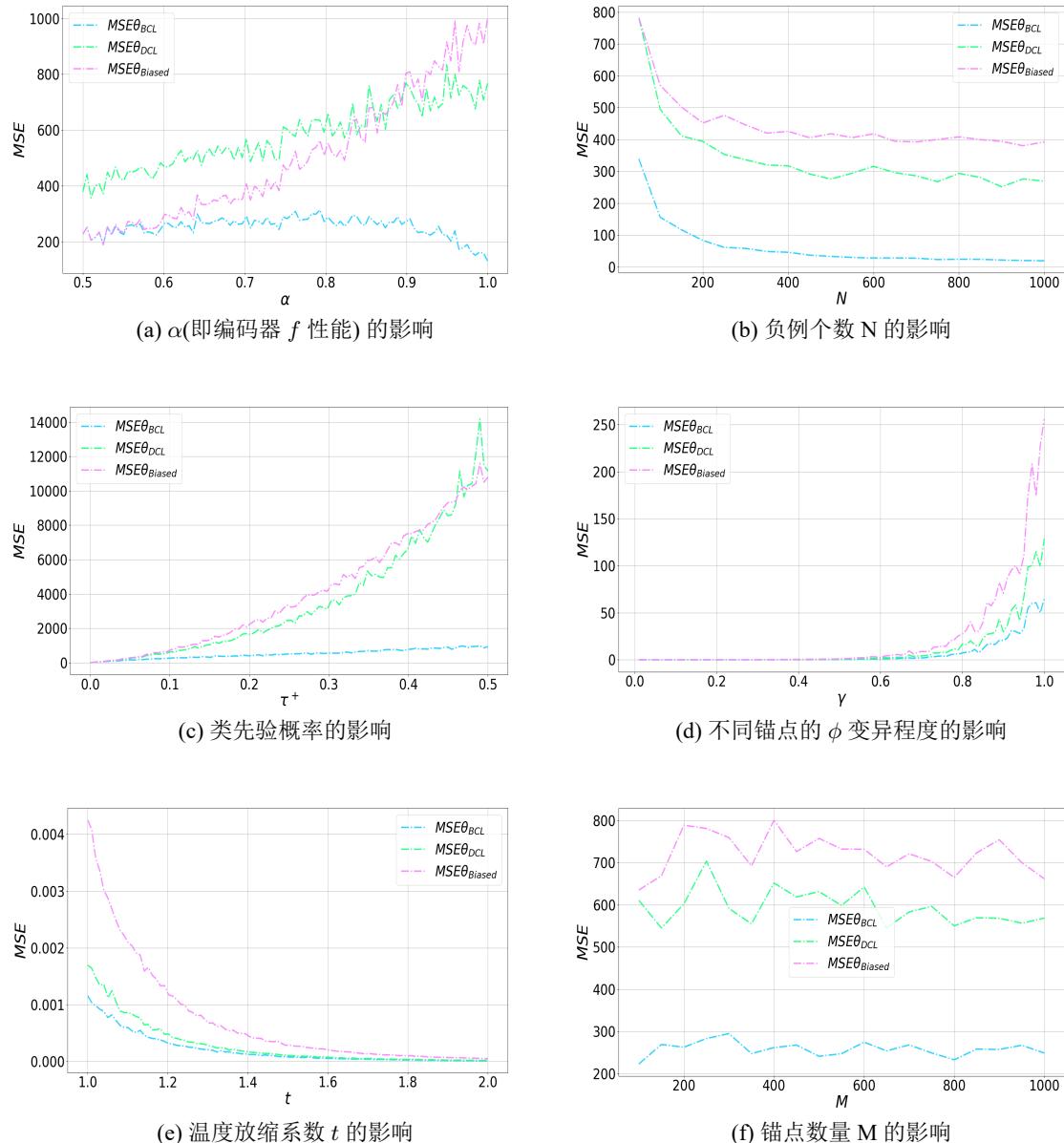


图 6.9 不同参数设置下估计量的均方误差比较

算估计量的均方误差:

$$MSE\hat{\theta} = \frac{1}{M} \sum_{m=1}^M (\hat{\theta}_m - \theta_{\text{SUP},m})^2$$

首先固定参数设置如下: $\alpha = 0.9$, $\gamma = 0.1$, $\tau^+ = 0.1$, 温度缩放 $t = 0.5$, 锚点数量 $M = 1e3$, 负样本数量 $N = 64$ 。然后变动不同的参数, 统计不同估计量的均方误差。图6.9对不同参数下不同估计量的均方误差 MSE 进行了比较。可以观察到, 在不同

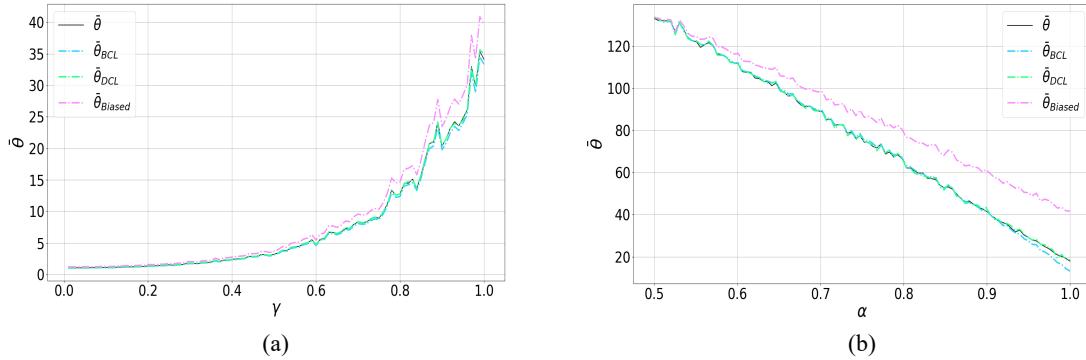


图 6.10 不同参数设置下估计量的均值比较

的参数设置下， $\hat{\theta}_{BCL}$ 估计量相对于其它估计量具有更低的均方误差。DCL 方法主要受制于额外正样本得个数，在实践中，通常只有非常有限得额外正例个数可供使用。需要重点关注的是，随着负例个数 N 的增大， $\hat{\theta}_{BCL}$ 的均方误差变小，与前面的分析一致。

除了 MSE 之外，图 6.10b 还报告了不同估计量 $\hat{\theta}$ 的均值在不同参数下的变化。受限于篇幅，重点关注分布 ϕ 或编码器 f 的选择是否会影响估计器的一致性。图 6.10a 显示了 ϕ 变异对均值的影响。图 6.10b 显示了编码器性能对均值的影响。可以看到， $\hat{\theta}_{BCL}$ 、 $\hat{\theta}_{DCL}$ 和 θ 序列的均值非常接近，这是由切比雪夫大数定律保证的，即，当 $\mathbb{E} \hat{\theta}_m = \theta_m$ 时， $\lim_{M \rightarrow \infty} \mathbb{P}\{|\frac{1}{M} \sum_{m=1}^M \hat{\theta}_m - \frac{1}{M} \sum_{m=1}^M \theta_m| < \epsilon\} = 1$ 。这个结论不受建议分布 ϕ 或编码器 f 的选择影响。 $\hat{\theta}_{BCL}$ 、 $\hat{\theta}_{DCL}$ 都是真负例分数均值的无偏估计量，这是重要性采样的性质以及均值估计量的性质决定的。此外，图 6.10b 也可以看作是训练过程的模拟：性能更好的编码器，即更高的 α ，将真负样本编码得与锚点不相似（相似度分数更小），将正例编码得与锚点更近（相似度分数更大）。因此，观察到真负例分数均值 $\bar{\theta}$ 的减少，对应于训练过程中损失值的减少，以及偏差 $|\bar{\theta}_{Biased} - \bar{\theta}|$ 的增加，因为随着 α 的增加， $\bar{\theta}_{Biased}$ 中包含的伪负样本得分更高。

6.6.2 真实数据集实验

6.6.2.1 数据集

在个性化推荐任务上，与第 5 章相同的五个公开数据集：MovieLens-100k，MovieLens-1M，Yahoo!-R3，Yelp2018 和 Gowalla。数据预处理方法也一样：前三个

数据集包含用户评分，按照^[1,45,46]的方法将所有有评分的物品转换为隐式反馈。对于每个数据集，随机将其中 20% 的数据作为测试数据，剩下的 80% 用于训练。数据集的统计信息可以参考第5章的实验部分。

在图像分类任务上，使用了四个公开数据集，CIFAR10，STL10，CIFAR100 和 tinyImageNet 数据集。简单介绍 CIFAR10 数据集，CIFAR-10 数据集包含 10 个类别的 60000 张尺寸为 32x32 的彩色图像，每个类别有 6000 张图像。其中，训练集包含 50000 张图像，测试集包含 10000 张图像。不同类别的训练样本完全互斥，且类比平衡。图6.11提供了 CIFAR10 数据集的十个类别及相应的随机样本概览。其余三个数据集与 CIFAR10 数据集类似，都是类比平衡。除了 STL10 数据集以外，其他数据集都是完全标注的，但在训练过程中，训练集的标签都不使用，即负例无监督，仅在评估的时候使用类别标签。这四个数据集的统计信息如下表：

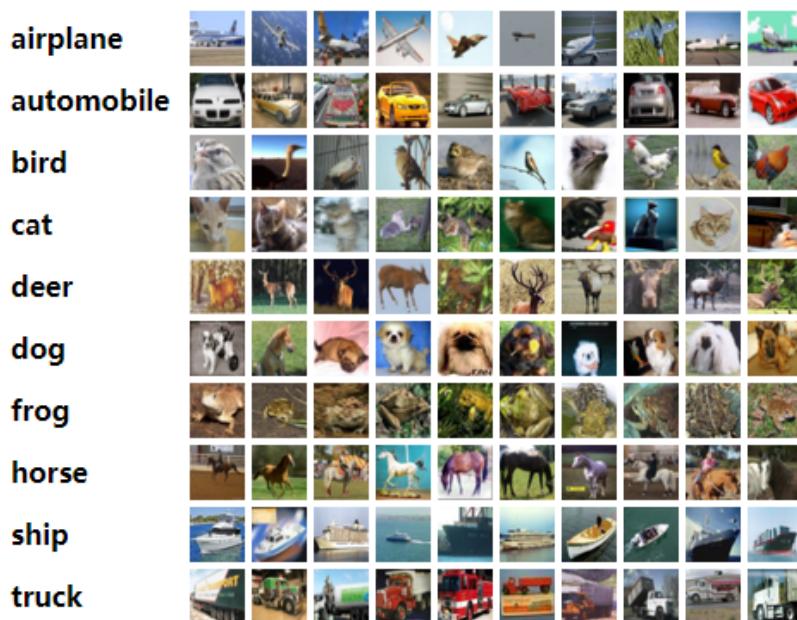


图 6.11 CIFAR10 数据集的十个类别及相应的随机样本概览

6.6.2.2 对比算法

由于本章的出发点是通过对负样本加权校准估计，从而校准负例的梯度，不适用于两个网络分支分别采用梯度法和动量法更新的异步网络架构，因此对比方法仅考虑以 LightGCN^[5] 和 SimCLR^[8] 为框架的同步法更新的网络架构，主要包含如下对比方法：

表 6.2 图像分类任务数据集统计信息

	类比总数	图像尺寸	训练集数量	验证集数量	测试集数量	是否完全标注
CIFAR10	10	32x32(RGB)	50,000	/	10,000	是
STL10	10	96x96(RGB)	105,000	/	8,000	否
CIFAR100	100	32x32(RGB)	50,000	/	10,000	是
tinyImageNet	200	64x64(RGB)	100,000	10,000	10,000	是

- InfoNCE^[7]: 具体表达式见式(6.2), 在负例无监督场景下, 该损失使用未标注样本的相似度分数替代真负例的相似度分数。
- DCL^[97]: 由于未标记数据中存在伪负例, DCL 提出了一个估计量见式(6.47)来修正 InfoNCE 损失分母中的第二项中真负例分数和的估计。
- HCL^[98]: 使用了 DCL 的去偏框架的基础上, 进一步使用每个随机选择的未标记样本的相似度分数加权进行困难样本挖掘。DCL 是 $\beta = 0$ 时 HCL 的一个特例。这些对比算法的含义在上一章实验部分已有介绍, 不再赘述。

6.6.2.3 实验设置

无论是个性化推荐任务还是图像分类任务, 都是通过优化对比损失学习样本的表示, 根据学到的表示进行下游排序或者分类任务, 因此学习样本的表示也被称为“代理任务”(pretext task)。尽管下游任务的评估协议略有差异: 推荐任务是根据用户表示的 K 近邻物品表示生成 top-K 推荐(即内积相似度高的物品), 而图像分类任务是将学到的表示固定, 然后训练一个线性分类器。但共同点是, 学到的对比表示质量越好, 即正例越对齐, 负例越均匀, 下游排序或者分类任务的泛化性能越好。

在推荐任务上, 使用两个编码器: 矩阵分解(MF)^[124] 和轻量级图卷积网络(LightGCN)^[5]。不同的推荐模型都会编码的用户和物品表示, 然后计算对比损失以训练推荐模型。根据学到的用户物品表示, 计算用户评分矩阵, 从而为每个用户预测个性化推荐列表。评估指标与前面保持一致, 包括精确度(P)、召回率(R)和归一化折损累积增益(NDCG), 其中 K 的取值为 5、10 和 20。

在图像分类任务方面进行公平的比较, 所有实验设置与 DCL^[97] 和 HCL^[98] 完全相同。具体地, 仅考虑同步同构的网络架构 SimCLR^[8] 框架, 它使用的是 InfoNCE 损失函数。不同的对比方法使用完全相同的数据增强方法, 使用 ResNet-50^[137] 作为编码器架构, 并使用学习率为 0.001 的 Adam 优化器^[138]。温度缩放系数和潜在向量

华中科技大学博士学位论文

的维度分别固定为 $t=0.5$ 和 $d=128$ ，所有模型训练 400 个 epochs。在评估上与推荐任务有所不同，图像分类任务的评估遵循已有的评估协议：首先固定学习到的嵌入，然后通过训练线性分类器进行评估^[98,139]，评估指标使用 ACC。在训练时均不使用标签，只在评估精度时使用样本标签。

6.6.3 实验结果

6.6.3.1 个性化推荐

表6.3提供了不同的学习算法在五个数据集上的 top-k 性能评估。含有去偏机制的 DCL 方法普遍由于有偏的 InfoNCE 方法，表明在 positive-unlabeled 的隐式反馈数据集上进行伪负例去偏是必要的，这个结论与前面章节结论一致。此外，包含硬负例挖掘机制的 HCL 方法和 BCL 方法普遍由于不含硬负例挖掘机制的 DCL 和 InfoNCE 方法，说明在隐式反馈数据集上进行硬负例挖掘的必要性。

需要着重强调的是 HCL 和 BCL 的差别，它们都包含困难负例挖掘和伪负例去偏机制，但是 BCL 显著优于 HCL 方法。这是由于两者重要性加权机制的不同所造成的：HCL 的权重可以简单理解为未标注样本相似度分数的指数次方，相似度分数越大，重要性权重越大，可以很好的兼顾硬负例挖掘任务，但无法兼顾伪负例去偏任务；BCL 的权重是未标注样本是真负例的后验概率，可以灵活地兼顾硬负例挖掘和伪负例去偏任务。此外，HCL 的权重是样本相似度分数的函数，而 BCL 的权重是经验分布函数(相对排序位置)的函数。在推荐中，嵌入倾向于过度平滑，从而导致相似度分数趋同，那么 HCL 的权重可能会消失。而 BCL 的权重不会受到相似度分数具体值的影响，因此表现出了更好的鲁棒性。

6.6.3.2 图像分类

前面讨论了，应该在显存可承受的范围内，设定尽量大的负例个数 N 值。以 24GB 显存的 RTX3090 显卡为例，能够承受的最大的 N=510，此时批量大小为 256(在 SimCLR 框架下，负例个数 N=2*batch size -2)。固定 N=510，然后使用不同的方法训练。其中，在 CIFRA10 数据集上， α 是通过经验评估计算的， $\beta = 0.9$ ；在 STL10 数据集上，设定 $\alpha = 0.75$ ， $\beta = 1.0$ ；在 STL10 数据集上，设定 $\alpha = 0.7$ ， $\beta = 1.0$ ；在 TinyImageNet 数据集上，设定 $\alpha = 0.8$ ， $\beta = 1.0$ ；可以看到，BCL 相对于对于 SimCLR，top-1 分类精度分别取得了 1.4%，6.3%，3.3% 和 3.9% 的绝对提升。相对于

表 6.3 Top-k 推荐性能比较

数据集	模型	学习算法	Top-5 评估			Top-10 评估			Top-20 评估		
			Precision	Recall	NDCG	Precision	Recall	NDCG	Precision	Recall	NDCG
MovieLens-100k	MF	Info_NCE	0.4081	0.1388	0.4324	0.3452	0.2266	0.4095	0.2793	0.3497	0.4118
		DCL	0.4168	0.1434	0.4458	0.3513	0.2291	0.4202	0.2835	0.3546	0.4207
		HCL	0.4263	0.1463	0.4539	0.3565	0.2323	0.426	0.2849	0.3564	0.4242
		BCL(Proposed)	0.4374	0.1552	0.4674	0.3658	0.2405	0.4380	0.2931	0.3588	0.4357
MovieLens-1M	LightGCN	Info_NCE	0.3924	0.1343	0.4209	0.3349	0.2183	0.4006	0.2679	0.3289	0.3976
		DCL	0.3962	0.1367	0.4243	0.3361	0.2194	0.4022	0.2695	0.3329	0.4006
		HCL	0.4197	0.1461	0.4501	0.3458	0.2256	0.4188	0.2802	0.3446	0.4182
		BCL(proposed)	0.4366	0.1512	0.4652	0.3626	0.2367	0.4351	0.2936	0.3599	0.4347
Yahoo!-R3	MF	InfoNCE	0.3820	0.0879	0.4003	0.3339	0.1478	0.3728	0.2821	0.2358	0.3605
		DCL	0.4009	0.0934	0.4209	0.3472	0.1546	0.3894	0.289	0.2423	0.3731
		HCL	0.4112	0.0969	0.4317	0.3552	0.1585	0.3991	0.2959	0.2475	0.3825
		BCL(proposed)	0.4249	0.1041	0.4439	0.3652	0.1657	0.4101	0.3012	0.2537	0.3920
Yelp2018	LightGCN	InfoNCE	0.4121	0.0986	0.4386	0.359	0.1594	0.4041	0.2979	0.2482	0.3869
		DCL	0.4104	0.0982	0.4291	0.3544	0.1597	0.3977	0.2965	0.2511	0.3842
		HCL	0.4107	0.0948	0.4300	0.3514	0.1542	0.3950	0.2916	0.2413	0.3775
		BCL(proposed)	0.4256	0.1047	0.4460	0.3651	0.1658	0.1893	0.2998	0.2534	0.3911
Gowalla	MF	Info_NCE	0.1429	0.1065	0.1615	0.1080	0.1601	0.1664	0.0786	0.2316	0.1952
		DCL	0.1454	0.1083	0.1635	0.1091	0.1618	0.1692	0.079	0.2327	0.1974
		HCL	0.1460	0.1097	0.1638	0.1096	0.1628	0.1697	0.0792	0.2336	0.1976
		BCL(proposed)	0.1529	0.1122	0.1685	0.1124	0.1667	0.1736	0.0815	0.2367	0.2029
Gowalla	LightGCN	Info_NCE	0.1417	0.1074	0.1676	0.1099	0.1633	0.1719	0.0798	0.2354	0.2007
		DCL	0.1456	0.1092	0.1642	0.1089	0.1622	0.1697	0.079	0.2333	0.1982
		HCL	0.1412	0.1139	0.1718	0.113	0.1683	0.1776	0.0812	0.2394	0.2059
		BCL(proposed)	0.1537	0.1140	0.1725	0.1156	0.1696	0.1771	0.0842	0.2428	0.2076
Gowalla	MF	Info_NCE	0.0429	0.0246	0.047	0.0365	0.0417	0.0491	0.0305	0.07	0.058
		DCL	0.0486	0.0278	0.0531	0.0410	0.0466	0.0552	0.0342	0.0777	0.0648
		HCL	0.0515	0.0305	0.0566	0.0459	0.0541	0.0622	0.0383	0.0894	0.0736
		BCL(proposed)	0.0546	0.0342	0.0595	0.0475	0.0563	0.0643	0.0398	0.0912	0.0752
Gowalla	LightGCN	Info_NCE	0.0553	0.0329	0.0607	0.0473	0.0558	0.0642	0.0390	0.0911	0.0754
		DCL	0.0559	0.0331	0.0612	0.0472	0.0557	0.0642	0.0391	0.0914	0.0756
		HCL	0.0563	0.0335	0.0617	0.0477	0.0564	0.0648	0.0393	0.0920	0.0760
		BCL(proposed)	0.0612	0.0376	0.0670	0.0524	0.0629	0.0701	0.0414	0.1014	0.0815
Gowalla	MF	Info_NCE	0.0739	0.0757	0.1016	0.0560	0.1122	0.1076	0.0422	0.1650	0.1230
		DCL	0.0746	0.0769	0.1023	0.0568	0.1147	0.1088	0.0426	0.1664	0.1238
		HCL	0.0755	0.0774	0.1035	0.0574	0.1151	0.1098	0.0432	0.1693	0.1256
		BCL(proposed)	0.0827	0.0837	0.1121	0.0639	0.1257	0.1186	0.0485	0.1824	0.1353
Gowalla	LightGCN	Info_NCE	0.0743	0.0760	0.1022	0.0566	0.1132	0.1084	0.0423	0.1649	0.1231
		DCL	0.0748	0.0763	0.1027	0.0569	0.1132	0.1088	0.0424	0.1656	0.1236
		HCL	0.0794	0.0804	0.1084	0.0608	0.1199	0.1147	0.0453	0.1740	0.1319
		BCL(proposed)	0.0863	0.0893	0.1156	0.0670	0.1332	0.1248	0.0490	0.1937	0.1420

最优对比方法 HCL, BCL 的 top-1 分类精度分别取得了 0.6%, 0.1%, 0.2% 和 0.3% 的绝对提升，在 CIFAR10 和 TinyImagenet 数据集上的提升具有显著性。

6.6.4 超参数分析

由于图像数据集上标注比较完备，而推荐数据集始终存在未标注样本，因此使用图像数据集分析不同超参数的影响。

表 6.4 Top-1 图像分类精度比较

数据集	负例个数 N	模型	SimCLR	DCL	HCL	BCL
CIFAR10	510	ResNet50	91.1	92.1	91.9	92.5 (±0.15)
STL10	510	ResNet50	80.2	84.3	87.4	87.5 (±0.12)
CIFAR100	510	ResNet50	66.4	67.7	69.5	69.7 (±0.10)
tinyImageNet	510	ResNet50	53.4	53.7	57.0	57.3 (±0.09)

负例个数 N 的影响: 首先固定 α 和 β 的值如上所述, 然后分析不同负例个数 N 对分类精度的影响。结果如表6.5所示, 对于所有的对比方法, 更大的负例个数 N 导致了分类精度一致的提升。这与对比损失本身的性质有关, 更大的 N 导致了更大的互信息下界。此外, 对于 HCL 方法而言, 更大的 N 导致 (1) 经验分布函数 $\hat{\Phi}_{UN}(\hat{x})$ 的估计误差减小; (2) α 的估计误差减小; (3) 重要性采样的估计误差减小, 从而使得 BCL 经验估计的估计误差确定性地减小, 这与数值实验部分的结论是一致的。

表 6.5 负例个数的影响

数据集	学习算法	负例个数 N				
		N=30	N=62	N=126	N=254	N=510
CIFAR10	SimCLR	80.21	84.82	87.58	89.87	91.12
	DCL	82.41	87.60	90.38	91.36	92.06
	HCL	83.42	88.45	90.53	91.57	91.92
	BCL	83.61	88.56	90.83	92.07	92.58
STL10	SimCLR	61.20	71.69	74.36	77.33	80.20
	DCL	63.91	71.48	76.69	81.48	84.26
	HCL	67.24	73.38	79.44	83.82	86.38
	BCL	67.45	73.36	80.23	84.68	86.51

 表 6.6 参数 β 的影响

Dataset	$\beta = 0.5$	$\beta = 0.6$	$\beta = 0.7$	$\beta = 0.8$	$\beta = 0.9$	$\beta = 1.0$
CIFAR10	91.39	91.41	91.89	92.02	92.58	92.12
STL10	80.32	81.79	83.58	83.83	84.85	86.51

β 的影响: α 对应于编码器的宏 AUC, 有明确的含义; 而 β 控制着采样分布的

难度级别 (hardness level)，是手动选择的，因此只分析参数 β 的影响。固定最优参数 α 不变，然后检查不同 β 值的影响。表 6.6 展示了难度级别 β 的影响。在 STL10 数据集上，随着 β 的增加，Top-1 分类准确率持续提高。然而，在 CIFAR10 数据集上，随着 β 的增加，Top-1 分类准确率逐渐增加，在 0.9 处达到最优值后开始下降。这结果表明，更高的难度级别并不总是导致更好的性能。这一观点也可以从 HCL 和 DCL 的性能比较得到印证，HCL（包含困难负样本挖掘机制）在 CIFAR10 数据集上的表现比 DCL（不包含困难负样本挖掘机制）要差。

如何设定一个合理的硬负例挖掘参数 β ？关于深度神经网络的记忆效应的研究^[140]为回答这个问题提供了启发。具体而言，已经证明，不论使用何种网络架构，深度模型会优先记忆带有干净标签的训练数据，并随着训练轮数的增加逐渐适应带有噪声标签的训练数据^[108,110]。自监督对比学习可以被视为一个负例含有噪声的噪声标签学习 (noisy label learning) 问题。较大的 β 值会导致给困难样本分配更高的权重，从而导致更大的损失值和相应的梯度值。为了避免神经网络拟合噪声，在正类先验较小的数据集（即噪声比较低的数据集）中，设置较大的 β 值可以加快对干净的困难样本的拟合，因此一个启发式的 β 设定值可以为

$$\beta = 1 - 1/C$$

其中， C 为数据集的类别总数，那么 $1/C$ 即为噪声率，越小的噪声率，那么应该设定一个更接近于 1 的 β 值。真是数据集的实验也验证了这个结论：在 100 类的 CIFAR100 和 200 类的 tinyImageNet 数据集上，最优的参数值为 1。除了噪声率以外，另外一个需要考虑的因素是数据集本身拟合的难易度水平。与 CIFAR-10 数据集相比，STL-10 数据集具有更大的挑战，它的 top-1 分类精度显著低于 CIFAR10。因此，在相对难以拟合的 STL-10 数据集中，应当使用较大的 β 值可以激励神经网络在拟合噪声样本之前先拟合干净的困难样本。在相对容易拟合的 CIFAR-10 数据集中，应当使用相对小一点的 $\beta = 0.9$ ，避免神经网络拟合噪声样本。

τ^+ 的影响：表 6.7 展示了正例类先验概率 τ^+ 的设置对分类精度的影响。结果显示最佳的 $\tau^+ = 1/C$ 。考虑数据集中的一个特定类别，比如“猫”，可以将属于猫类的图像数量记为随机变量 X ，它可以看作 N 次独立成功率为 τ^+ 伯努利试验的结果，那么 $X \sim B(Y, \tau^+)$ ，其中 Y 是数据集中的所有样本数。在这种情况下， τ^+ 的无偏估计 X/Y 。在一个平衡的数据集中，数据集中的总样本数为 $Y = C * X$ ，其中 C 是

表 6.7 正例类先验概率 τ^+ 的影响

数据集	N	τ^+	top1	top5
tinyImageNet	510	1e-3	57.10	80.54
tinyImageNet	510	5e-3	57.32	80.87
tinyImageNet	510	1e-2	56.34	80.24

类别的数量， X 是每个类别中的样本数，于是有 $\tau^+ = 1/C$ 。

6.6.5 运行时间比较

前面分析了，本章的方法的额外计算开销主要是由于经验分布函数的计算，其复杂度为 $O(N)$ ，其中 $N = 2 * (\text{batch size}-1)$ ，相比于编码一个样本，这个开销是可以忽略的。表6.8比较了在不同数据集上进行单个训练周期的实际运行时间。其中，批量大小固定为 256，编码器架构为 ResNet-50。在 CIFAR10、STL10 和 CIFAR100 数据集上的实验是在配备了 Intel(R) Xeon(R) Platinum 8255C CPU 和单卡的 RTX 3090(24GB) * 1 GPU 的云服务器上进行的。在 tinyImageNet 数据集上的实验是在配备了 Intel(R) Xeon(R) Platinum 8352M CPU 和单卡的 A100-SXM4-80GB(80GB) * 1 GPU 的云服务器上进行的。结果显示，本章提出的方法几乎相比于对比方法有着相同的运行时间。tinyImageNet 数据集运行时间较长是因为图像增强模块中，随机剪裁后图像的尺寸相对于前面三个数据集更大。

表 6.8 算法运行时间比较

数据集	Batch Size	SimCLR	BCL
CIFAR10	256	82s	82s
STL10	256	169s	169s
CIFAR100	256	82s	82s
tinyImageNet	256	237s	238s

6.7 本章小结

本章提出贝叶斯自监督对比学习 (BCL) 算法，通过重要性权重来纠正从无标签数据中随机选择的负样本导致的偏差。从单个权重来看，BCL 提供了样本是真负例的后验概率估计。BCL 计算的权重是样本是真负例的后验概率线性函数，为真负样

华 中 科 技 大 学 博 士 学 位 论 文

本赋以较大权值，遵循硬负例原则；为伪负例赋以较小的权值，遵循真负例原则。从损失函数值来看，BCL 提供了有监督数据下对比损失的一致估计。通过重要性加权未标注样本相似度分数，实现对真负例相似度分数的近似，间接地实现了对有监督损失的近似，从而提升下游任务的泛化性能。其关键思想是在贝叶斯框架下推导目标采样群体的概率，该分布具有参数化结构，可以灵活统一地执行硬负例挖掘和伪负例去偏任务这两个冲突任务。在数值实验、个性化推荐、图像分类任务上验证了贝叶斯自监督对比学习 (BCL) 算法的有效性，只需要简单地修改损失函数，而无需额外的存储和计算开销。

本章对前面章节方法的局限性进行了全面的考量并改进：针对第三章正例弱监督问题，本章解决了更具普遍性的负例无监督问题，且解决方案不再依赖于表示向量高斯分布的参数化假设，从而成功地将模型从简单的矩阵分解向图卷积神经网络和残差网络进行了推广。针对第四章显式负采样带来的额外计算开销问题，本章通过校正估计重写损失函数不引入额外的计算开销，与基线方法有着相同的运行时间。此外通过图6.3对正例和负例相似度分数的分析，突破了第四章所依赖的“正例分数大于负例分数”的假设，获取类条件概率密度不再依赖任何参数化的假设。针对第五章仅聚焦于单个负例的对比损失去偏，本章通过重要性权重加权将去偏目标向更具一般性的多个负例的对比损失推广；此外在伪负例去偏的基础上，进一步考虑了硬负例挖掘，综合全面地考虑了对比学习的任务需求。

7 总结与展望

7.1 全文总结

搜索推荐场景面临较为突出的正例弱监督、负例无监督的问题，使得从推荐数据集中学习良好的样本特征面临严重的挑战，并对排序预测任务产生了不利影响。对比学习利用了数据本身的结构和特性，无需额外的标注信息，就可以从数据中学到有用的特征表示，是自监督学习的一种重要的实现方式，也是提升推荐算法准确性的有效途径。本文从推荐场景中正例弱监督、负例无监督的问题出发，针对自监督对比学习的三个关键组件正例、负例、损失函数做出了改进，提出了相应的正例去噪算法、负例采样算法和损失函数校正算法，并且将算法从只有一个负例的成对学习向更具一般性的多个负例的对比学习推广。本文的创新和贡献如下：

针对正例弱监督引发的伪正样本问题，提出了自适应学习的成对排序算法，用于从含噪成对比较的数据中学习个性化排序。分析了隐式反馈弱监督的本质是数据不完全 (incomplete data)，并把从含噪成对比较的数据中学习排序的问题形式化为一个含有隐变量的最大后验估计问题。引入一个隐变量作为衡量交互置信度的新指标，这个隐变量与用户物品的特征表示一起作为模型参数一起端到端地学习。所提出的算法采用期望最大化框架：在期望步骤中使用贝叶斯推断来估计交互的置信度指标；在最大化步骤中固定置信度指标，更新参数学习用户和物品的特征表示。该算法在合成的噪声数据集上表现出了更好的鲁棒性，在真实的数据集上表现出更高的推荐准确性。

针对负例无监督引发的伪负样本问题，提出了贝叶斯最优负采样算法，用于从未标记的数据中采样高质量的负例，以提升对比学习的训练效果并提升推荐精度。提出了贝叶斯最优负采样准则，这是最小化经验采样风险的理论最优的采样规则。该方法指定了后验概率意义上的负信号测度，结合了静态的先验信息和动态的样本信息，统一了现有的两种提取负信号的两种主要范式，并为使用辅助信息建模先验概率的方法提供了灵活的接口。该算法在采样误差率和采样样本信息量以及推荐准确性上优于同类方法。

针对负例无监督导致的有偏成对损失函数，提出了去偏成对学习算法，以近似

完全监督数据的成对损失函数，以获得接近于有监督学习的泛化性能。该方法通过校正伪负样本导致的概率估计偏差，从而修正梯度以近似完全监督数据的梯度。所提出的目标函数易于实现，不需要额外的辅助信息进行监督，也不需要过多的存储和计算开销。在保持严格的相对于成对学习的线性时间复杂度情况下，去偏成对学习算法取得了更高的推荐准确性。

针对负例无监督导致的有偏对比损失函数，提出了贝叶斯自监督对比学习算法，通过重要性权重来纠正从无标签数据中随机选择的负样本引入的偏差。从单个权重来看，该方法提供了样本是真负例的后验概率估计，可以灵活统一地执行硬负例挖掘和伪负例去偏任务这两个冲突任务。从损失函数值来看，该方法提供了有监督数据下对比损失的一致估计，从而提升下游任务的泛化性能。在数值实验、个性化推荐和图像分类等多个任务上验证了贝叶斯自监督对比学习算法的有效性。只需要简单地修改损失函数，而无需额外的存储和计算开销。

7.2 后续工作展望

针对推荐数据集弱监督问题，本文围绕正例、负例和损失函数这三个对比学习的关键组件展开研究，并提出了正例去噪算法、负例采样算法和损失函数校正算法。然而，从弱监督的隐式反馈数据集中学习良好的样本特征仍存在一些值得关注的问题，从参数更新方式、图数据增强方式和网络架构设计三个方面总结如下：

在参数更新方式方面，以 MOCO、BYOL 为代表的模型采用异步更新策略，即分别采用梯度法和动量法更新两个网络分支参数。然而，为何动量法的更新策略有效，仍是一个自监督对比学习尚需回答的问题。本文的损失函数校正虽然针对同步更新策略设计，但实质上通过校正损失函数来调节梯度，改变参数更新方式。深刻理解动量法工作机制，将基于梯度法的参数更新改进方式推广至动量法，设计更优的动量更新策略以提升自监督对比学习效果，是自监督对比学习领域一个值得探索的重要理论和实践问题。

在数据增强方面，获取可靠的图数据增强是解决正例噪声的有效手段，比本文第三章提出的含有隐变量的最大后验估计更高效。由于图数据的非欧几里得特性，难以像图像数据那样通过随机剪裁、高斯模糊等操作获得语义不变的数据增强。在数据增强方法方面，目前主流的基于随机过程的图数据增强可能会损失图中重要的

华 中 科 技 大 学 博 士 学 位 论 文

结构信息，如何获取稳定、可靠且保留重要结构信息的增强图，是一个重要的问题。从对比的方式来看，推荐中主流的方式是将原图和增强图进行对比，即只进行一次数据增强。如何组织互补的图数据增强方式，使得增强图之间进行相互对比，从而受益于更密集的数据增强，是提升推荐性能的重要途径。

在网络架构设计方面，采用异步异构的网络架构设计无需负样本，有效地避免了伪负例的错误梯度和网络坍塌，取得了令人惊艳的结果。在推荐中，样本特征提取所依赖的图卷积网络不同于提取图像数据的残差网络，具有结构上的特殊性。如何设计符合推荐场景的异步更新策略和网络架构，是一个值得探索的问题。

致 谢

对在课题研究及论文写作过程中给予指导和帮助的导师、校内外专家、实验技术人员、同学等表示感谢。

在致谢时建议具体，不同的人如何助力完成你的论文，都需要特别注明。如导师、其他老师或实验技术人员、以及同学对你论文的贡献是不一样的，有指引课题方向、修改论文，也有具体教会实验操作，也有协助你做了哪方向的实验，或者给你精神安慰、陪你度过紧张的研究生生涯。

越具体越能表达你真实的感受，否则就是毫无意义的套话。

参考文献

- [1] S. Rendle, C. Freudenthaler, Z. Gantner, L. Schmidt-Thieme. BPR: Bayesian Personalized Ranking from Implicit Feedback. in: UAI 2009, Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, June 18-21, 2009, 2009:452-461
- [2] Q. Ai, K. Bi, J. Guo, W. B. Croft. Learning a deep listwise context model for ranking refinement. in: The 41st international ACM SIGIR conference on research & development in information retrieval, 2018:135–144
- [3] J. Ding, Y. Quan, Q. Yao, Y. Li, D. Jin. Simplify and Robustify Negative Sampling for Implicit Collaborative Filtering. in: Proceedings of the International Conference on Neural Information Processing Systems, 2020
- [4] J. Ding, Y. Quan, X. He, Y. Li, D. Jin. Reinforced Negative Sampling for Recommendation with Exposure Data. in: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, 2019:2230–2236
- [5] H. Xiangnan, D. Kuan, W. Xiang, L. Yan, Z. Yongdong, W. Meng. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation. in: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, 2020:10
- [6] X. Wang, X. He, M. Wang, F. Feng, T. S. Chua. Neural Graph Collaborative Filtering. in: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019:2344-2353
- [7] A. v. d. Oord, Y. Li, O. Vinyals. Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748, 2018
- [8] T. Chen, S. Kornblith, M. Norouzi, G. Hinton. A Simple Framework for Contrastive Learning of Visual Representations. in: International conference on machine learning, 2020:1597–1607
- [9] K. He, H. Fan, Y. Wu, S. Xie, R. Girshick. Momentum contrast for unsupervised visual representation learning. in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020:9729–9738
- [10] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, et al. Bootstrap Your Own Latent a New Approach to Self-Supervised Learning. in: Proceedings of the 34th International Conference on Neural Information Processing Systems, 2020:21271-21284

华中科技大学博士学位论文

- [11] R. A. Bradley, M. E. Terry. Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons. *Biometrika*, 1952, 39(3/4):324-345
- [12] R. D. Luce. Individual Choice Behavior: A Theoretical Analysis. *Journal of the American Statistical Association*, 2005, 67(293):1-15
- [13] W. Pan, L. Chen. Gbpr: Group preference based bayesian personalized ranking for one-class collaborative filtering. in: Twenty-Third International Joint Conference on Artificial Intelligence, 2013
- [14] R. Yu, Y. Zhang, Y. Ye, L. Wu, C. Wang, Q. Liu, et al. Multiple pairwise ranking with implicit feedback. in: Proceedings of the 27th ACM Int. Conference on Information and Knowledge Management, ACM, 2018:1727-1730
- [15] X. Jiang, L.-H. Lim, Y. Yao, Y. Ye. Statistical ranking and combinatorial Hodge theory. *Mathematical Programming*, 2011, 127(1):203-244
- [16] X. Yang, B. Wang. Local ranking and global fusion for personalized recommendation. *Applied Soft Computing*, 2020, 96(1):106636
- [17] H. Qiu, Y. Liu, G. Guo, Z. Sun, J. Zhang, H. T. Nguyen. BPRH: Bayesian personalized ranking for heterogeneous implicit feedback. *Information Sciences*, 2018, 453:80-98
- [18] F. Zhao, Y. Shen, X. Gui, J. H. SDBPR: Social distance-aware bayesian personalized ranking for recommendation. *Future Generation Computer Systems*, 2019, 95:372-381
- [19] L. Maystre, M. Grossglauser. Just Sort It! A Simple and Effective Approach to Active Preference Learning. in: Proceedings of the 34th International Conference on Machine Learning, 2017:2344-2353
- [20] X. Yang, B. Wang. Destructure-and-restructure matrix approximation. *Information Sciences*, 2020, 514(2020):434-448
- [21] D. McFadden. Conditional logit analysis of qualitative choice behavior. *Frontiers in Econometrics*, 1974
- [22] C. Dwork, R. Kumar, M. Naor, D. Sivakumar. Rank aggregation methods for the Web. in: Proceedings of the 10th international conference on World Wide Web, 2001:613-622
- [23] K. J. Arrow. Social Choice and Individual Values. Yale University Press, 2012
- [24] M. Gutmann, A. Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. in: Proceedings of the thirteenth international conference on artificial intelligence and statistics, 2010:297–304

华中科技大学博士学位论文

- [25] M. U. Gutmann, A. Hyvärinen. Noise-Contrastive Estimation of Unnormalized Statistical Models, with Applications to Natural Image Statistics. *Journal of machine learning research*, 2012, 13(2)
- [26] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, et al. Self-supervised learning: Generative or contrastive. *IEEE Transactions on Knowledge and Data Engineering*, 2021
- [27] J. Wu, X. Wang, X. Gao, J. Chen, H. Fu, T. Q. He. On the Effectiveness of Sampled Softmax Loss for Item Recommendation. *CoRR*, 2022, abs/2201.02327
- [28] X. Cai, C. Huang, L. Xia, X. Ren. LightGCL: Simple Yet Effective Graph Contrastive Learning for Recommendation. *arXiv preprint arXiv:2302.08191*, 2023
- [29] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018
- [30] A. Dosovitskiy, J. T. Springenberg, M. Riedmiller, T. Brox. Discriminative unsupervised feature learning with convolutional neural networks. in: *Proceedings of the International Conference on Neural Information Processing Systems*, 2014:766–774
- [31] F. Wenqi, L. Xiaorui, J. Wei, Z. Xiangyu, T. Jiliang, L. Qing. Graph Trend Networks for Recommendations. in: *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2021:12
- [32] T. Wang, P. Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. in: *International conference on machine learning*, 2020:9929–9939
- [33] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, et al. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018
- [34] N. Saunshi, O. Plevrakis, S. Arora, M. Khodak, H. Khandeparkar. A theoretical analysis of contrastive unsupervised representation learning. in: *International conference on machine learning*, 2019:5628–5637
- [35] O. Henaff. Data-efficient image recognition with contrastive predictive coding. in: *International conference on machine learning*, 2020:4182–4192
- [36] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, et al. Supervised contrastive learning. in: *Proceedings of the International Conference on Neural Information Processing Systems*, 2020:18661–18673
- [37] P. Bachman, R. D. Hjelm, W. Buchwalter. Learning representations by maximizing mutual information across views. 2019

华中科技大学博士学位论文

- [38] X. Chen, H. Fan, R. Girshick, K. He. Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297, 2020
- [39] J. Huang, Q. Dong, S. Gong, X. Zhu. Unsupervised deep learning by neighbourhood discovery. in: International conference on machine learning, 2019:2849–2858
- [40] Z. Wu, Y. Xiong, S. X. Yu, D. Lin. Unsupervised feature learning via non-parametric instance discrimination. in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2018:3733–3742
- [41] C. Zhuang, A. L. Zhai, D. Yamins. Local aggregation for unsupervised learning of visual embeddings. in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019:6002–6012
- [42] X. Ren, L. Xia, J. Zhao, D. Yin, C. Huang. Disentangled Contrastive Collaborative Filtering. arXiv preprint arXiv:2305.02759, 2023
- [43] W. He, G. Sun, J. Lu, X. S. Fang. Candidate-aware Graph Contrastive Learning for Recommendation. in: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2023:1670–1679
- [44] Y. Yang, Z. Wu, L. Wu, K. Zhang, R. Hong, Z. Zhang, et al. Generative-Contrastive Graph Learning for Recommendation. in: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2023
- [45] S. Rendle, C. Freudenthaler. Improving pairwise learning for item recommendation from implicit feedback. in: Proceedings of the 7th ACM international conference on Web Search and Data Mining, 2014:273-282
- [46] W. Zhang, T. Chen, J. Wang, Y. Yu. Optimizing Top-n Collaborative Filtering via Dynamic Negative Item Sampling. in: Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2013:785–788
- [47] T. Zhao, J. McAuley, I. King. Improving Latent Factor Models via Personalized Feature Projection for One Class Recommendation. in: Proceedings of the 24th ACM International Conference on Information and Knowledge Management, 2015:821–830
- [48] B. Liu, B. Wang. Bayesian Negative Sampling for Recommendation. in: 2023 IEEE 39th International Conference on Data Engineering (ICDE). IEEE, 2023:749–761
- [49] W. Shi, J. Chen, F. Feng, J. Zhang, J. Wu, C. Gao, et al. On the Theories Behind Hard Negative Sampling for Recommendation. in: Proceedings of the ACM Web Conference 2023, 2023:812–822

华中科技大学博士学位论文

- [50] X. Qin, N. Sheikh, B. Reinwald, L. Wu. Relation-aware Graph Attention Model With Adaptive Self-adversarial Training. in: Proceedings of the AAAI Conference on Artificial Intelligence, 2021:9368–9376
- [51] H. Zhao, X. Yang, Z. Wang, E. Yang, C. Deng. Graph debiased contrastive learning with joint representation clustering. in: Proceedings of the International Joint Conference on Artificial Intelligence, 2021:3434–3440
- [52] T. Chen, Y. Sun, Y. Shi, L. Hong. On Sampling Strategies for Neural Network-Based Collaborative Filtering. in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017:767–776
- [53] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean. Distributed Representations of Words and Phrases and Their Compositionality. in: Proceedings of the 26th International Conference on Neural Information Processing Systems, 2013:3111–3119
- [54] C. Wang, Y. Yu, W. Ma, M. Zhang, C. Chen, Y. Liu, et al. Towards Representation Alignment and Uniformity in Collaborative Filtering. in: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2022:1816–1825
- [55] G. Zhu, W. Lu, C. Yuan, Y. Huang. AdaMCL: Adaptive Fusion Multi-View Contrastive Learning for Collaborative Filtering. in: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2023:1076–1085
- [56] X. Qin, H. Yuan, P. Zhao, J. Fang, F. Zhuang, G. Liu, et al. Meta-optimized Contrastive Learning for Sequential Recommendation. arXiv preprint arXiv:2304.07763, 2023
- [57] J. Shuai, K. Zhang, L. Wu, P. Sun, R. Hong, M. Wang, et al. A review-aware graph contrastive learning framework for recommendation. in: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2022:1283–1293
- [58] 李航. 统计学习方法. 清华大学出版社, 2019
- [59] D. H. Park, Y. Chang. Adversarial Sampling and Training for Semi-Supervised Information Retrieval. in: World Wide Web Conference, 2019:1443–1453
- [60] T. Huang, Y. Dong, M. Ding, Z. Yang, W. Feng, X. Wang, et al. MixGCF: An Improved Training Method for Graph Neural Network-Based Recommender Systems. in: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021:665–674

华中科技大学博士学位论文

- [61] J. Ding, Y. Quan, X. He, Y. Li, D. Jin. Reinforced Negative Sampling for Recommendation with Exposure Data. in: Proceedings of the International Joint Conference on Artificial Intelligence, 2019:2230–2236
- [62] Z. Yang, M. Ding, C. Zhou, H. Yang, J. Zhou, J. Tang. Understanding negative sampling in graph representation learning. in: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020:1666–1676
- [63] X. Wang, Y. Xu, X. He, Y. Cao, M. Wang, Chua. Reinforced negative sampling over knowledge graph for recommendation. in: Proceedings of International Conference on World Wide Web, 2020:99–109
- [64] Y. Wang, Z. Liu, Z. Fan, L. Sun, P. S. Yu. Dskreg: Differentiable sampling on knowledge graph for recommendation with relational gnn. in: Proceedings of the 30th ACM International Conference on Information & Knowledge Management, 2021:3513–3517
- [65] J. Chen, C. Wang, S. Zhou, Q. Shi, Y. Feng, C. Chen. Samwalker: Social recommendation with informative sampling strategy. in: Proceedings International Conference on World Wide Web, 2019:228–239
- [66] C. Wang, J. Chen, S. Zhou, Q. Shi, Y. Feng, C. Chen. SamWalker++: recommendation with informative sampling strategy. IEEE Transactions on Knowledge and Data Engineering, 2021
- [67] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, J. Leskovec. Graph Convolutional Neural Networks for Web-Scale Recommender Systems. in: Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2018:974–983
- [68] T. Zhao, J. McAuley, I. King. Leveraging social connections to improve personalized ranking for collaborative filtering. in: Proceedings of the 23rd ACM international conference on conference on information and knowledge management, 2014:261–270
- [69] X. Wang, W. Lu, M. Ester, C. Wang, C. Chen. Social recommendation with strong and weak ties. in: Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, 2016:5–14
- [70] F. Yuan, J. M. Jose, G. Guo, L. Chen, H. Yu, R. S. Alkhawaldeh. Joint geo-spatial preference and pairwise ranking for point-of-interest recommendation. in: 2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI), 2016:46–53

华中科技大学博士学位论文

- [71] W. Liu, Z.-J. Wang, B. Yao, J. Yin. Geo-ALM: POI Recommendation by Fusing Geographical Information and Adversarial Learning Mechanism. in: Proceedings of the International Joint Conference on Artificial Intelligence, 2019:1807–1813
- [72] J. Ding, F. Feng, X. He, G. Yu, Y. Li, D. Jin. An Improved Sampler for Bayesian Personalized Ranking by Leveraging View Data. in: Companion Proceedings of the The Web Conference 2018, 2018:13–14
- [73] J. Wang, L. Yu, W. Zhang, Y. Gong, Y. Xu, B. Wang, et al. IRGAN: A Minimax Game for Unifying Generative and Discriminative Information Retrieval Models. in: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, 2017:515–524
- [74] R. Zhu, B. Zhao, J. Liu, Z. Sun, C. W. Chen. Improving contrastive learning by visualizing feature transformation. in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021:10306–10315
- [75] Z. Zhong, E. Fini, S. Roy, Z. Luo, E. Ricci, N. Sebe. Neighborhood contrastive learning for novel class discovery. in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021:10867–10875
- [76] N. Komodakis, S. Gidaris. Unsupervised representation learning by predicting image rotations. in: ICLR, 2018
- [77] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, et al. Going deeper with convolutions. in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2015:1–9
- [78] S. Liu, R. Ying, H. Dong, L. Li, T. Xu, Y. Rong, et al. Local augmentation for graph neural networks. in: International Conference on Machine Learning. PMLR, 2022:14054–14072
- [79] P. Veličković, W. Fedus, W. L. Hamilton, P. Liò, Y. Bengio, R. D. Hjelm. Deep graph infomax. arXiv preprint arXiv:1809.10341, 2018
- [80] Y. Zhu, Y. Xu, F. Yu, Q. Liu, S. Wu, L. Wang. Deep graph contrastive representation learning. arXiv preprint arXiv:2006.04131, 2020
- [81] Y. You, T. Chen, Y. Sui, T. Chen, Z. Wang, Y. Shen. Graph contrastive learning with augmentations. Advances in neural information processing systems, 2020, 33:5812–5823
- [82] Y. You, T. Chen, Y. Shen, Z. Wang. Graph Contrastive Learning Automated. in: Proceedings of the 38th International Conference on Machine Learning, 2021:12121–12132

华中科技大学博士学位论文

- [83] C. Zheng, B. Zong, W. Cheng, D. Song, J. Ni, W. Yu, et al. Robust Graph Representation Learning via Neural Sparsification. in: Proceedings of the 37th International Conference on Machine Learning, 2020:11458–11468
- [84] D. Luo, W. Cheng, W. Yu, B. Zong, J. Ni, H. Chen, et al. Learning to Drop: Robust Graph Neural Network via Topological Denoising. in: Proceedings of the 14th ACM International Conference on Web Search and Data Mining, 2021:779–787
- [85] T. Zhao, X. Zhang, S. Wang. GraphSMOTE: Imbalanced Node Classification on Graphs with Graph Neural Networks. in: Proceedings of the 14th ACM International Conference on Web Search and Data Mining, 2021:833–841
- [86] L. Page. The pagerank citation ranking: Bringing order to the web. Technical report. Stanford Digital Library Technologies Project, 1998, 1998
- [87] J. Zhang, H. Zhang, C. Xia, L. Sun. Graph-bert: Only attention is needed for learning graph representations. arXiv preprint arXiv:2001.05140, 2020
- [88] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, et al. Manifold mixup: Better representations by interpolating hidden states. in: International conference on machine learning. PMLR, 2019:6438–6447
- [89] L. Lerche, D. Jannach. Using graded implicit feedback for bayesian personalized ranking. in: Proceedings of the 8th ACM Conference on Recommender systems, 2014:353–356
- [90] W. Yu, Z. Qin. Spectrum-enhanced Pairwise Learning to Rank. in: World Wide Web Conference, 2019:2247–2257
- [91] B. Wu, Y. Ye. BSPR: Basket-sensitive personalized ranking for product recommendation. Information Sciences, 2020, 541(1):185–206
- [92] C. Huang, S. Wang, X. Wang, L. Yao. Dual Contrastive Transformer for Hierarchical Preference Modeling in Sequential Recommendation. in: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2023:99–109
- [93] L. Jing, P. Vincent, Y. LeCun, Y. Tian. Understanding dimensional collapse in contrastive self-supervised learning. International Conference on Learning Representations, 2021
- [94] C.-H. Yeh, C.-Y. Hong, Y.-C. Hsu, T.-L. Liu, Y. Chen, Y. LeCun. Decoupled contrastive learning. in: European Conference on Computer Vision. Springer, 2022:668–684
- [95] J. Li, P. Zhou, C. Xiong, S. C. H. Hoi. Prototypical Contrastive Learning of Unsupervised Representations

华中科技大学博士学位论文

- [96] Z. Lin, C. Tian, Y. Hou, W. X. Zhao. Improving graph collaborative filtering with neighborhood-enriched contrastive learning. in: Proceedings of the ACM Web Conference 2022, 2022:2320–2329
- [97] C.-Y. Chuang, J. Robinson, Y.-C. Lin, A. Torralba, S. Jegelka. Debiased Contrastive Learning. in: Proceedings of the International Conference on Neural Information Processing Systems, 2020:8765–8775
- [98] J. Robinson, C. Ching-Yao, S. Sra, S. Jegelka. Contrastive Learning with Hard Negative Samples. in: ICLR, 2021
- [99] 范诗松, 王静龙, 潘晓龙. 高等数理统计. 高等教育出版社, 2006
- [100] V. Glivenko. Sulla determinazione empirica delle leggi di probabilità. Gion. Ist. Ital. Attauri., 1933, 4:92–99
- [101] H. A. David, H. N. Nagaraja. Order statistics. 2004
- [102] 范诗松. 贝叶斯统计. 贝叶斯统计, 1999
- [103] A. K. Menon, B. Van Rooyen, N. Natarajan. Learning from binary labels with instance-dependent noise. Machine Learning, 2018, 107:1561–1595
- [104] G. Patrini, A. Rozza, A. Krishna Menon, R. Nock, L. Qu. Making deep neural networks robust to label noise: A loss correction approach. in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017:1944–1952
- [105] B. Van Rooyen, R. C. Williamson. A Theory of Learning with Corrupted Labels. in: JMLR, 2017
- [106] X. Xia, T. Liu, N. Wang, B. Han, C. Gong, G. Niu, et al. Are anchor points really indispensable in label-noise learning? in: Advances in neural information processing systems, 2019
- [107] T. Liu, D. Tao. Classification with noisy labels by importance reweighting. IEEE Transactions on pattern analysis and machine intelligence, 2015, 38(3):447–461
- [108] C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Vinyals. Understanding deep learning (still) requires rethinking generalization. Communications of the ACM, 2021, 64(3):107–115
- [109] M. Li, M. Soltanolkotabi, S. Oymak. Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks. in: International conference on artificial intelligence and statistics. PMLR, 2020:4313–4324
- [110] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, et al. Co-teaching: Robust training of deep neural networks with extremely noisy labels. in: Advances in neural information processing systems, 2018:31

华中科技大学博士学位论文

- [111] D. B. Madan, E. Seneta. The variance gamma (VG) model for share market returns. *Journal of business*, 1990, 41(A):511-524
- [112] E. Seneta. Fitting the variance-gamma model to financial data. *Journal of applied probability*, 2004, 41(A):177-187
- [113] A. P. Dempster, N. M. Laird, D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 1977, 39(1):1-22
- [114] W. Wang, F. Feng, X. He, L. Nie, T.-S. Chua. Denoising implicit feedback for recommendation. in: *Proceedings of the 14th ACM international conference on web search and data mining*, 2021:373–381
- [115] 周志华. 机器学习. 航空港, 2018
- [116] B. Jessa, D. Jesse. Learning from positive and unlabeled data: a survey. *Machine Learning*, 2020, 109:719–760
- [117] G. Su, W. Chen, M. Xu. Positive-unlabeled learning from imbalanced data. in: *Proceedings of the 30th International Joint Conference on Artificial Intelligence, Virtual Event*, 2021:2995–3001
- [118] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, Q. Mei. LINE: Large-Scale Information Network Embedding. in: *Proceedings International Conference on World Wide Web*, 2015:1067–1077
- [119] L. Xu, J. Lian, W. X. Zhao, M. Gong, L. Shou, D. Jiang, et al. Negative Sampling for Contrastive Representation Learning: A Review. *arXiv preprint arXiv:2206.00212*, 2022
- [120] A. Herschtal, B. Raskutti. Optimising area under the ROC curve using gradient descent. in: *Proceedings of the twenty-first International conference on Machine learning*, 2004:49
- [121] X. He, H. Zhang, M.-Y. Kan, T.-S. Chua. Fast matrix factorization for online recommendation with implicit feedback. in: *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, 549–558:2016
- [122] Z. Zhang, B. Wang. Graph Neighborhood Routing and Random Walk for Session-based Recommendation. in: *IEEE International Conference on Data Mining (ICDM)*, 2021:1517-1522
- [123] 陶哲轩. 陶哲轩实分析. 人民邮电出版社, 2018
- [124] Y. Koren, R. Bell, C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 2009, 42(8):30–37

华中科技大学博士学位论文

- [125] F. Wang, H. Liu. Understanding the Behaviour of Contrastive Loss. in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021:2495–2504
- [126] S. Jain, M. White, P. Radivojac. Estimating the class prior and posterior from noisy positives and unlabeled data. in: Advances in Neural Information Processing Systems, 2016
- [127] M. Christoffel, G. Niu, M. Sugiyama. Class-prior estimation for learning from positive and unlabeled data. in: Asian Conference on Machine Learning, 2016:221–236
- [128] J. Ding, Y. Quan, Q. Yao, Y. Li, D. Jin. Simplify and Robustify Negative Sampling for Implicit Collaborative Filtering. in: Proceedings of the International Conference on Neural Information Processing Systems, 2019
- [129] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, G. E. Hinton. Big self-supervised models are strong semi-supervised learners. Proceedings of the International Conference on Neural Information Processing Systems, 2020, 33:22243–22255
- [130] B. Poole, S. Ozair, A. Oord, A. A. Alemi, G. Tucker. On Variational Bounds of Mutual Information. in: International conference on machine learning, 2019:5171–5180
- [131] M. I. Belghazi, A. Baratin, S. Rajeswar, S. Ozair, Y. Bengio, A. Courville, et al. MINE:Mutual Information Neural Estimation. in: International conference on machine learning, 2018
- [132] J. Xia, L. Wu, G. Wang, J. Chen, S. Z. Li. ProGCL: Rethinking Hard Negative Mining in Graph Contrastive Learning. in: International conference on machine learning, 2022:24332–24346
- [133] B. Lindsay. Mixture Models: Theory, Geometry and Applications. 1995
- [134] T. C. Hesterberg. Advances in importance sampling. Stanford University, 1988
- [135] Y. Bengio, J.-S. Senécal. Adaptive importance sampling to accelerate training of a neural probabilistic language model. IEEE Transactions on Neural Networks, 2008, 19(4):713–722
- [136] G. Casella, C. P. Robert, M. T. Wells. Generalized accept-reject sampling schemes. Lecture Notes-Monograph Series, 2004, : 342–347
- [137] K. He, X. Zhang, S. Ren, J. Sun. Deep residual learning for image recognition. in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2016:770–778
- [138] P. Diederik, B. Jimmy. Adam: A method for stochastic optimization. in: ICLR, 2015

华 中 科 技 大 学 博 士 学 位 论 文

- [139] L. Lajanugen, L. Honglak. An efficient framework for learning sentence representations. in: ICLR, 2018
- [140] D. Arpit, S. Jastrzkebski, N. Ballas, D. Krueger, E. Bengio, M. S. Kanwal, et al. A closer look at memorization in deep networks. in: International conference on machine learning, 2017:233–242

附录 1 答辩委员会决议

首行缩进两个字符，中文字体采用小四宋体，英文字体采用 Times New Roman，
字体大小为小四，行间距为固定值 20 磅。

附录 2 攻读博士学位期间取得的研究成果

发表与接收论文

- [1] **Bin Liu**, Bang Wang. Bayesian Negative Sampling for Recommendation, 39th IEEE International Conference on Data Engineering (ICDE), 2023. (CCF-A; 第一署名单位: 华中科技大学)
- [2] **Liu Bin**, Wang, Bang. Pairwise Learning for Personalized Ranking with Noisy Comparisons. Information Sciences, 2022. (SCI 一区; IF=8.10; 第一署名单位: 华中科技大学)
- [3] **Bin Liu**, Tao Chen, Peng Jia, Lin Wang. Effective public service delivery supported by time-decayed Bayesian personalized ranking. Knowledge-Based Systems, 2020. (SCI 一区; IF=8.80; 第一署名单位: 华中科技大学)
- [4] **Bin Liu**, Bang Wang. Bayesian self-supervised Contrastive Learning, 2023. (under review; arXiv:2301.11673; 第一署名单位: 华中科技大学)
- [5] **Bin Liu**, Qin Luo, Bang Wang. Debiased Pairwise Learning from Positive-Unlabeled Implicit Feedback, 2023. (under review; arXiv:2307.15973; 第一署名单位: 华中科技大学)
- [6] **Bin Liu**, Eerjia Chen, Bang Wang. Reducing Popularity Bias in Recommender Systems through AUC-Optimal Negative Sampling, 2023. (under review; arXiv:2306.01348; 第一署名单位: 华中科技大学)

科技奖励

- [1] 全部作者的姓名全称, 本人的名字加粗. 题目. 国家级/省部级科技类奖, 获奖年
- [2] 全部作者的姓名全称, 本人的名字加粗. 题目. 国际/国内竞赛类奖, 获奖年

附录 3 公开发表的学术论文与博士学位论文的关系

序号	成果名称	成果形式	成果主要内容	与学位论文对应的关系
1				
2				
3				
4				
5				

附录 4 攻读博士学位期间参与的科研项目

1. 项目类型

项目名称: 项目名称

项目编号: No. 88888888

起止时间: 2018 年 8 月至 2018 年 8 月

担任角色: 担任角色