

面向推荐系统中有偏和无偏一元反馈建模的 三任务变分自编码器

林子楠^{1,2,3}, 刘杜钢^{1,2,3}, 潘微科^{1,2,3}, 明 仲^{1,2,3}

¹ 深圳大学大数据系统计算技术国家工程实验室 深圳 中国 518060

² 人工智能与数字经济广东省实验室(深圳) 深圳 中国 518060

³ 深圳大学计算机与软件学院 深圳 中国 518060

摘要 一元反馈建模在推荐系统中的应用非常广泛,例如点击预测和购买预测等。然而,推荐系统作为一个闭环的反馈系统,在用户与系统的交互过程中可能存在着多种偏置问题,例如位置偏置、流行偏置等,进而导致用户的反馈数据存在有偏性。现有的大部分推荐模型都只基于这样的有偏数据进行构建,忽略了偏置的影响,进而导致推荐结果是次优的。目前已有的偏置消除方法大致可以分为基于反事实学习的方法、基于启发式的方法和基于无偏数据增强的方法。其中,基于无偏数据增强的方法通常被认为在稳定性和准确性方面表现较好。

本文重点研究了推荐系统中一元反馈的偏置问题,通过引入由一种特定策略收集的无偏数据,结合有偏数据进行联合建模,从而得到更准确和无偏的推荐模型。具体而言,本文从多任务学习的角度对问题进行建模,将有偏数据、无偏数据以及它们的并集当作三种相互关联的信号,并设计了三个不同但相关的学习任务。变分自编码器是目前最先进的一元反馈建模方法之一,有着独特的建模方式,从而使得它在很多问题中取得了优越的推荐效果。本文基于变分自编码器提出了一种新的推荐模型,即三任务变分自编码器(Tri-VAE)。该模型包含三个变分自编码器,分别对三种信号进行重构。三个变分自编码器之间共享同一个编码器和同一个解码器。此外,本文还设计了特征校正模块和标签增强模块以加强任务之间的关联。其中,特征校正模块用来校正用户的潜在特征,得到更无偏的潜在特征,进而从潜在特征的角度缓解偏置的影响。标签增强模块用于生成可靠性较高的伪标签并加以利用,进而更有效地利用无偏数据中的信息。

在 Yahoo! R3 和 Coat Shopping 两个公开数据集上的实验结果表明,所提出的模型相比于目前最新的基线模型在绝大多数情况下取得了显著的效果提升。为了进一步研究所提出的模型,本文进行了消融实验、超参数敏感性分析和收敛性分析,还对特征校正模块的有效性进行了探讨。

关键词 偏置问题; 联合建模; 多任务学习; 变分自编码器; 推荐系统

中图分类号 TP181 DOI号 10.19363/J.cnki.cn10-1380/tn.2021.09.09

Tri-task Variational Autoencoder for Modeling of Biased and Unbiased Unary Feedback in Recommender Systems

LIN Zinan^{1,2,3}, LIU Dugang^{1,2,3}, PAN Weike^{1,2,3}, MING Zhong^{1,2,3}

¹ National Engineering Laboratory for Big Data System Computing Technology, Shenzhen University, Shenzhen 518060, China

² Guangdong Laboratory for Artificial Intelligence and Digital Economy(SZ), Shenzhen 518060, China

³ College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China

Abstract Modeling of unary feedback is widely used in recommender systems for click prediction and purchase prediction. However, as a closed-loop feedback system, a recommender system may have various bias issues during the user-system interaction, such as position bias and popularity bias, leading to biased user feedback data. Most of the existing recommendation models are constructed based on such biased data, ignoring the influence of the bias, which leads to sub-optimal recommendation results. The existing works for bias elimination can be roughly divided into three categories, i.e., counterfactual learning-based methods, heuristic-based methods and unbiased data augmentation methods. Among them, the unbiased data augmentation methods are usually considered to be of better stability and accuracy.

In this paper, we focus on the bias problem of unary feedback in recommender systems. By introducing an unbiased data collected by a specific policy and then jointly modeling it with the biased data, we are able to learn a more accurate and unbiased recommendation model. Specifically, from the perspective of multi-task learning, we regard the biased data, the unbiased data and their union as three correlated signals, and then design three different but related preference learning tasks. Variational autoencoder is a state-of-the-art method for modeling unary feedback. Because of its unique modeling

通讯作者: 潘微科, 博士, 副教授, Email: panweike@szu.edu.cn

本课题得到国家自然科学基金重点项目(No. 61836005)和国家自然科学基金面上项目(No. 61872249)资助。

收稿日期: 2021-04-26; 修改日期: 2021-07-30; 定稿日期: 2021-08-10

ideas, it has achieved superior recommendation performance in many cases. Based on variational autoencoder, we design a novel recommendation model called tri-task variational autoencoder (Tri-VAE). Our proposed model contains three variational autoencoders that reconstruct three different signals, respectively. All variational autoencoders share the same encoder and the same decoder. In addition, we also design a feature calibration module and a label enhancement module to strengthen the association between the three tasks. In particular, the feature calibration module is used to calibrate the latent features of users to obtain more unbiased latent features, which in turn mitigates the impact of bias from the perspective of latent features. The label enhancement module is implemented to generate pseudo labels with high reliability, which can be exploited to make our proposed model more effective in exploiting the information in unbiased data.

The experimental results on two public datasets, including Yahoo! R3 and Coat Shopping, show that the proposed model achieves significantly better performance compared with the latest baseline models in most cases. In order to further study our proposed model, we conduct ablation studies, hyper-parameter sensitivity analysis and convergence analysis and discuss the effectiveness of the feature calibration module.

Key words bias issue; joint modeling; multi-task learning; variational autoencoder; recommender system

1 引言

随着互联网的快速发展,信息的数量呈现出爆炸性的增长,这进一步加剧了信息过载的问题。为了缓解这一问题,推荐系统应运而生^[1-4],并被用于从海量的信息中为用户提供有效的信息,从而满足用户的需求。推荐系统的基本思路是通过用户对用户的历史行为、兴趣偏好等信息进行建模和分析,挖掘出用户的需求,进而为用户推荐可能感兴趣的物品。

在推荐系统中,用户的行为(即对物品的反馈)可分为多元反馈、二元反馈和一元反馈等。多元反馈指的是用户对物品的反馈是显式且具有多个等级的,例如用户给物品进行打分,打分范围为{1,2,3,4,5}。二元反馈和多元反馈类似,用户对物品的反馈也是显式的,但不同于多元反馈,用户的反馈只有两个等级,即喜欢和不喜欢。而一元反馈指的是用户对物品的反馈是隐式的,是由系统根据用户的行为自行记录的,例如点击和购买等。在现实生活中,多元反馈和二元反馈都需要用户主动对物品进行反馈,收集过程较为困难,而一元反馈是由系统自行记录的,收集过程相对容易,这使得一元反馈在推荐系统中更为常见,并且得到了更广泛的应用。然而,一元反馈建模存在一个巨大的挑战,即用户的反馈不一定能反映用户的真实偏好,比如点击并不代表用户一定喜欢,而没有反馈也不代表用户不感兴趣,有可能只是物品没有被用户观测到,从而使得没有浏览、点击或购买等行为被系统记录下来。

现有的大部分推荐模型在建模的过程中,往往忽略了一个重要问题,即推荐系统中的偏置问题。因为推荐系统作为一个闭环的反馈系统,在用户与系统的交互过程中容易出现多种偏置问题^[5-8]。例如位置偏置,它指的是用户倾向于点击展示位置靠前的物品^[6];流行偏置,它指的是系统倾向于向用户推荐流行度较高的物品,也就是越受欢迎的物品越可能

被推荐给用户^[7]。除此之外,还有选择偏置、曝光偏置等等,这些偏置使得系统收集到的用户历史行为数据存在有偏性,而现有的大部分推荐模型都只基于这样的有偏数据来构建。一些研究工作从理论和实验两方面说明,忽略偏置问题会导致得到次优的推荐模型,而解决这些偏置问题能够提高推荐模型的性能^[5,9-11]。因此,基于这样的背景,本文重点研究了推荐系统中一元反馈的偏置问题,旨在提出一种新的推荐模型。

为了缓解这一偏置问题,本文引入了无偏数据,通过与有偏数据进行联合建模,来获得更加精准的推荐模型。其中,无偏数据指的是推荐系统通过采用随机的推荐策略而收集到的反馈数据,因为推荐策略的随机性,故可以被认为是不受偏置影响的^[10-12]。有偏数据指的是推荐系统通过采用常规的推荐策略而收集到的反馈数据,被认为是固有地受到偏置影响的。一般而言,因为采用随机策略的推荐结果较差,会带来用户的流失和平台的损失,所以无偏数据的收集规模通常都较小。因此,如何有效地利用无偏数据是本文研究问题的挑战之一。除此之外,有偏数据的规模比无偏数据大得多,且由于推荐策略的不同导致有偏数据和无偏数据之间存在异构性,如何有效地结合有偏数据和无偏数据进行联合建模是本文研究问题的另一个挑战。

先前的研究工作表明引入无偏数据可以有效缓解偏置的影响^[10-11]。在文献[10]中,作者提出了一种新的领域自适应算法,通过正则化约束来纠正模型的参数,从而达到消除偏置的目的。但是,当模型参数维度较高时,参数对齐较难训练,使得模型不能取得很好的效果。在文献[11]中,作者提出了一个新的基于知识蒸馏的框架,并从四个不同层面提出了多种策略,旨在通过利用无偏数据或由无偏数据训练的预训练模型来消除偏置,但是作者只是简单地使用了某一种策略,不能充分利用无偏数据中的信

息。除此之外, 先前的大部分研究工作认为偏置隐式地反映在标签或者模型参数上, 并且只在标签或模型参数的层面上来缓解偏置问题^[5,9-13], 很少有研究工作从特征的角度来解决偏置问题^[14]。

因此, 根据研究问题的挑战和已有研究工作的局限, 本文提出了一种新的视角, 即从多任务学习的角度来解决偏置问题, 把有偏数据、无偏数据和联合数据当作三种相互关联的信号, 分别对应三个不同但相关的学习任务。其中, 联合数据指的是有偏数据和无偏数据的并集。目前在一元反馈问题中, 变分自编码器模型取得了非常好的推荐性能^[15-16]。为此, 本文基于变分自编码器提出了一个新的模型, 即三任务变分自编码器。该模型包含三个变分自编码器, 分别对这三种数据进行重构。变分自编码器之间共享同一个编码器和同一个解码器。此外, 在三个任务之间, 本文设计了特征校正模块和标签增强模块来加强任务之间的关联。其中, 特征校正模块用于校正用户的潜在特征, 得到更无偏的潜在特征, 进而从潜在特征的角度来缓解偏置的影响。标签增强模块通过无偏数据训练的无偏模型来生成可靠性较高的伪标签并加以利用, 从而更有效地利用无偏数据中的信息。

本文的主要贡献如下:

1) 研究了推荐系统中的一个重要问题——一元反馈的偏置问题, 提出了一种新的解决方案, 将结合有偏数据和无偏数据的偏置消除问题建模成一个多任务学习问题, 即把有偏数据、无偏数据和联合数据当作三种不同但相互关联的数据, 并为此设计了一个新的三任务变分自编码器;

2) 提出了特征校正模块和标签增强模块, 其中, 特征校正模块用于获得无偏的用户潜在特征, 进而从潜在特征的角度缓解偏置的影响, 而标签增强模块能够更加充分地利用无偏数据中的信息;

3) 在两个不同领域的真实数据集上进行了大量的实验, 结果表明, 相比于现有最新的解决偏置的推荐模型, 本文提出的模型取得了更好的推荐效果。

2 相关工作

2.1 面向推荐系统中一元反馈建模的方法

面向推荐系统中一元反馈建模的方法指的是通过一元反馈数据来学习用户和物品之间的关系, 为每个用户生成对应的个性化推荐列表。到目前为止, 推荐系统中一元反馈方法大致可以划分为基于邻域的方法、基于分解的方法和基于神经网络的方法。

基于邻域的方法包括基于用户邻域的方法和基

于物品邻域的方法。基于用户邻域的方法通过计算用户之间的相似度, 为每个用户找到最相似的用户集合, 并推荐与相似用户有交互而与用户本身未交互的物品^[17-19]; 类似地, 基于物品邻域的方法则是计算物品之间的相似度, 为每个用户找到和他自身有交互的物品的相似物品, 并进行推荐^[20-21]。常用的计算相似度的方法有余弦相似度、杰卡德指数和关联规则^[22]等等。此类算法计算简单、可解释性较好, 但无法学习用户与物品之间较为复杂的关系。

基于分解的方法的主要思想是构建低维度的数学模型来表示用户与物品之间的关系, 再通过训练数据来学习模型的参数。比较有代表性的工作有贝叶斯个性化排序^[23]、分解的物品相似度模型(Factored Item Similarity Models, FISM)^[24]和对数几率矩阵分解(Logistic Matrix Factorization, LMF)^[25]。贝叶斯个性化排序采用成对的思想来构建形如三元组 (u, i, j) 的训练数据, 并认为用户 u 对 i 的喜欢程度大于 j , 其中 u 表示用户, i 表示与用户 u 有交互的物品, j 表示与用户 u 没有交互的物品。结合基于矩阵分解的思想, 即使用两个维度更小的矩阵来表示用户和物品的潜在特征, 通过使用三元组数据进行训练, 学习用户和物品的潜在特征。分解的物品相似度模型(FISM)创新性地提出物品之间的相似度可以通过学习得到, 并通过构建物品的潜在特征向量, 利用训练数据来学习用户对物品的偏好程度。类似地, 对数几率矩阵分解(LMF)结合矩阵分解的思想, 并利用对数几率损失函数来拟合数据, 学习用户和物品的潜在特征。此类方法能够有效学习用户与物品之间的线性关系, 但无法学习它们之间更为复杂的非线性关系。

基于神经网络的方法在近几年才被提出, 通过利用深度学习技术来学习用户与物品之间的高维度关系, 弥补了以往算法无法学习它们之间非线性关系的缺陷^[16,26-27]。此类方法包括基于多层感知器的方法和基于自编码器的方法, 其中比较有代表性的工作有神经协同过滤(Neural Collaborative Filtering, NCF)^[26]和变分自编码器(Variational Autoencoder, VAE)^[16]等。神经协同过滤(NCF)使用多层感知器代替原本用户和物品潜在特征之间的点乘操作, 从而实现非线性交互。此外, NCF将矩阵分解和多层感知器统一在同一个框架下, 结合了它们之间的优势, 同时捕获用户和物品的潜在特征之间的线性和非线性交互。变分自编码器(VAE)是目前在一元反馈建模方法中推荐性能最好的方法之一。它首先将用户与物品的交互记录编码成变分概率分布, 并使用多项

式似然来模拟一元反馈数据的生成过程,对原始输入数据进行重构。区别于以往的研究工作,变分自编码器学到的是用户潜在特征的概率分布,而不是用户潜在特征的点值,这也是该算法取得非常突出的性能的主要原因。

2.2 面向推荐系统中有偏反馈建模的方法

推荐系统中偏置消除问题起步较晚,近几年才逐渐被学术界和工业界重视。目前偏置消除的方法大致可以分为基于反事实学习的方法、基于启发式的方法和基于无偏数据增强的方法^[11]。

基于反事实学习的方法主要是从因果推理的角度来缓解偏置问题,目前其中一个主流的方法是引入逆倾向性得分(Inverse Propensity Score, IPS)评估器,即给每一个观测到的样本都赋予一个倾向性得分,从而学习一个更无偏的模型^[9,28-29]。此类方法需要根据一些假设来设计倾向得分评估器,因为真实的倾向性得分是无法求得的,从而导致此类方法的准确性和稳定性取决于评估方法本身。在文献[28]中,作者通过引入曝光概率,根据物品的流行度来计算每个样本的倾向性得分。在文献[29]中,作者通过神经网络模型来学习样本的倾向性得分。但是,这些倾向性得分的评估方法容易出现高方差等缺点,为了缓解这一问题,在文献[30]中,作者提出了双稳健(Doubly Robust, DR)评估器,利用插补模型来提高模型的稳定性。除此之外,在文献[13]中,作者提出了一个非对称训练框架,利用两个模型来产生更具无偏性的伪标签,以此加强第三个模型的训练。但该方法的有效性取决于伪标签的准确性,而伪标签的生成又受限于超参数的设置。在文献[31]中,作者建立了有关流行偏置产生的因果图,认为用户对物品的反馈和用户与物品之间的匹配程度、物品的流行度和用户的活跃度有关,并为此设计了一个多任务学习框架。但该方法仅用于解决流行偏置问题,具有一定的局限性。

基于启发式的方法目前较少,其主要思路是通过分析造成偏置的原因并作适当的假设,以此达到消除偏置的目的。但是偏置往往较为复杂,并不是由单一因素造成的,所以采用此类方法来解决偏置也是比较困难的。在文献[32]中,作者假设推荐系统中偏置的产生和沉默的螺旋理论有关,并提出了一种模拟沉默螺旋产生的概率模型。

基于无偏数据增强的方法通过引入无偏数据,对有偏数据和无偏数据进行联合建模,从而达到解决偏置问题的目的^[10-11,33-35]。此类方法是目前稳定性和准确性较好的方法,因为与上述方法相比,此类

方法引入了更多真实的无偏信息。在文献[34-35]中,作者利用无偏数据来协助倾向性得分的计算,得到更加稳定且准确的倾向性得分。同样地,此类方法也受限于倾向性得分评估方法的准确性。在文献[10]中,作者认为偏置的产生可以通过模型参数来进行纠正并提出了 CauseE 算法,通过引入额外的正则化项来约束有偏模型和无偏模型的参数。其中,有偏模型是指使用有偏数据进行训练得到的模型,而无偏模型是指使用无偏数据进行训练得到的模型。然而,当模型参数的维度较高时,参数对齐本身也是一个较难的问题。在文献[11]中,作者提出了一个基于知识蒸馏的框架,即利用无偏数据训练得到一个无偏模型,从蒸馏的角度提取无偏模型中的有效信息,进而和有偏数据进行联合训练来消除偏置的影响。但是,该方法没有充分利用无偏数据中蕴含的信息,因为仅仅是简单地使用了某一种策略。在文献[33]中,作者先利用无偏数据训练得到因果迁移随机森林模型,接着再利用有偏数据和无偏数据进行校正。该方法主要用于解决位置偏置问题,且模型复杂度较高。因此,为了更好地利用无偏数据中的信息,本文从多任务学习的角度出发,联合有偏数据和无偏数据进行建模,并设计了特征校正模块和标签增强模块来缓解偏置问题。

3 问题描述

3.1 定义

本文使用 $\mathcal{U} = \{u\} = \{1, 2, \dots, n\}$ 表示所有用户的集合, $\mathcal{I} = \{i\} = \{1, 2, \dots, m\}$ 表示所有物品的集合, $S^A = \{(u, i)\}$ 和 $S^T = \{(u, i)\}$ 表示包含这些用户对物品的一元反馈的两个日志数据。其中, S^A 是系统使用常规的推荐策略进行推荐时产生的,具有规模较大和相对有偏的特点; S^T 是系统使用随机的推荐策略进行推荐时产生的,具有规模较小和相对无偏的特点。本文使用 $X^A \in \{0, 1\}^{n \times m}$ 和 $X^T \in \{0, 1\}^{n \times m}$ 分别表示由 S^A 和 S^T 构成的用户与物品的交互矩阵。此外,本文使用 S^C 表示联合数据,即 $S^C = S^A \cup S^T$,以及 $X^C \in \{0, 1\}^{n \times m}$ 表示由 S^C 构成的用户与物品的交互矩阵。本文的目标是在给定有偏数据 S^A 和无偏数据 S^T 的前提下,如何对它们进行联合建模来缓解偏置问题,从而获得一个更加精准的个性化推荐模型。

3.2 挑战

本文研究的问题存在以下三个挑战:

1) 通常情况下, 无偏数据的规模较小。为了尽可能地减少偏置的影响, 从推荐系统中收集无偏数据需要采用随机的推荐策略。然而, 采用随机策略的推荐结果往往很差, 这不仅会降低用户的体验, 而且会导致系统带来的收益大幅下降。因此, 为了尽可能地减少损失, 无偏数据的收集过程不会持续很久, 从而使得其数据规模是较为有限的。

2) 有偏数据和无偏数据的规模差别较大, 有着严重的不平衡性。不同于无偏数据, 获取有偏数据是相对容易的, 从而导致了它们之间的数据规模的不一致性。

3) 有偏数据和无偏数据具有异构性, 因为它们分别是由两组不同性质的推荐策略产生的, 导致用户对物品的反馈所代表的偏好信息有所不同。在有偏数据中, 用户对物品的反馈不一定是因为用户喜欢物品, 有可能是受到了偏置的影响; 而在无偏数据中, 这通常表示用户喜欢相应的物品。

4 三任务变分自编码器

4.1 概述

针对上述挑战, 本文从一个新的角度对有偏数据和无偏数据进行联合建模, 将有偏数据 S^A 、无偏数据 S^T 和联合数据 S^C 看作 3 种不同但相互关联的

信号, 分别用于学习不同的用户偏好, 并基于变分自编码器, 提出了一种新的推荐模型, 即三任务变分自编码器(Tri-VAE), 如图 1 所示。Tri-VAE 包含了 3 个子模型, 分别记为 M^A 、 M^T 和 M^C 。它们的作用分别是对 S^A 、 S^T 和 S^C 进行重构。在这个 3 个子模型中, M^C 作为目标子模型, 其预测输出作为最终的推荐结果, 而另外两个子模型作为辅助子模型。3 个子模型之间通过以下两个方面进行关联, 一是它们共享同一个编码器和同一个解码器, 二是 M^A 和 M^T 通过特征校正模块(Feature Calibration Module, FCM)和标签增强模块(Label Enhancement Module, LEM)来协助 M^C 的学习。其中, 特征校正模块指的是利用 M^A 和 M^T 的用户潜在特征来校正 M^C 的用户潜在特征, 以获得更无偏的用户潜在特征; 标签增强模块则是进一步利用 M^T 的预测输出来训练 M^C , 因为 M^T 是重构无偏数据, 可以认为是相对无偏的模型, 其预测输出包含了较为丰富的无偏信息。

4.2 有偏数据重构

在 Tri-VAE 中, 子模型 M^A 的作用是重构有偏数据 S^A 。该子模型是变分自编码器, 其模型结构和 Mult-VAE^[16]一样, 包含编码器和解码器。其中, 编码器是推理模型, 将输入数据通过多层感知器压缩成

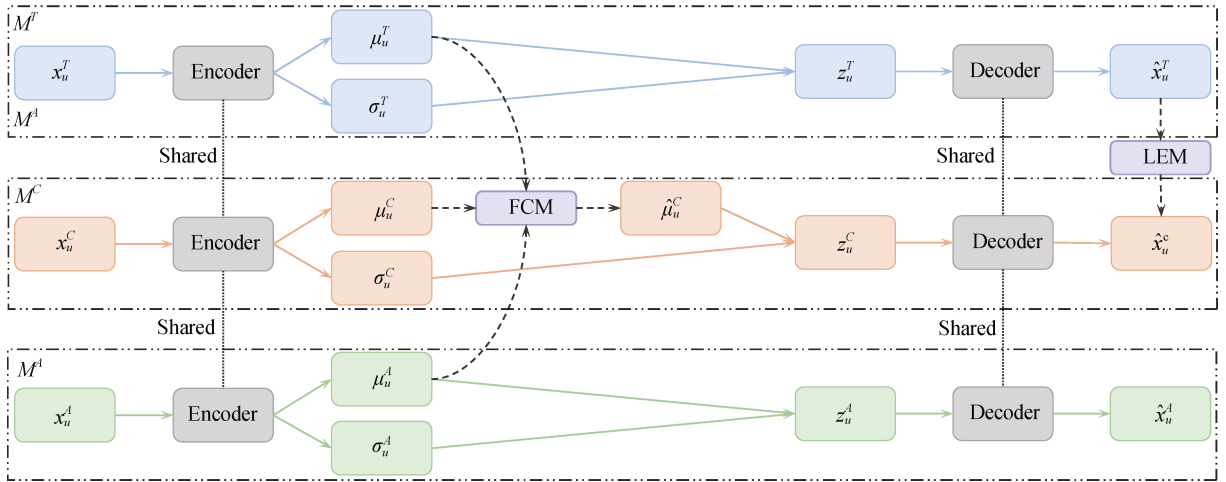


图 1 三任务变分自编码器模型架构。该模型包含三个子模型, 分别是 M^A 、 M^T 和 M^C , 其中 M^C 作为目标子模型, 旨在产生更精准的推荐结果。三个子模型之间共享同一个编码器和同一个解码器, 并通过特征校正模块 (Feature Calibration Module, FCM) 和标签增强模块 (Label Enhancement Module, LEM) 进行关联

Figure 1 Illustration of tri-task variational autoencoder (Tri-VAE). It contains three sub-models of M^A , M^T and M^C , where M^C is used as the target sub-model to produce more accurate recommendation results. The three sub-models share the same encoder and decoder, and are associated through the feature calibration module (FCM) and the label enhancement module (LEM).

用户潜在特征的变分分布, 得到均值和标准差; 解码器则是生成模型, 从用户潜在特征的变分分布中随机采样, 通过多层感知器对原始输入数据进行重构。在子模型 M^A 中, 使用 $\mathbf{z}_u^A \in \mathbb{R}^{1 \times k}$ 表示用户潜在特征, $\boldsymbol{\mu}_u^A \in \mathbb{R}^{1 \times k}$ 和 $\boldsymbol{\sigma}_u^A \in \mathbb{R}^{1 \times k}$ 分别表示用户潜在特征的变分分布的均值和标准差, 其中 k 指的是用户潜在特征的维度。由 \mathbf{X}^A 可以得到每个用户与所有物品产生的交互向量, 记为 $\mathbf{x}_u^A \in \{0,1\}^{1 \times m}$, 并将其作为编码器的输入, 通过多层感知器可以得到用户潜在特征的变分分布 $q_\phi(\mathbf{z}_u^A | \mathbf{x}_u^A)$, 从该变分分布随机采样可得到 \mathbf{z}_u^A , 进而有

$$q_\phi(\mathbf{z}_u^A | \mathbf{x}_u^A) = \mathcal{N}(\boldsymbol{\mu}_u^A, \text{diag}((\boldsymbol{\sigma}_u^A)^2)) \quad (1)$$

$$\boldsymbol{\mu}_u^A = \text{ReLU}(\mathbf{x}_u^A \mathbf{w}_\mu + \mathbf{b}_\mu) \quad (2)$$

$$\boldsymbol{\sigma}_u^A = \exp(\mathbf{x}_u^A \mathbf{w}_\sigma + \mathbf{b}_\sigma) \quad (3)$$

其中, $\mathbf{w}_\mu \in \mathbb{R}^{m \times k}$ 和 $\mathbf{w}_\sigma \in \mathbb{R}^{m \times k}$ 指的是权重矩阵, $\mathbf{b}_\mu \in \mathbb{R}^{1 \times k}$ 和 $\mathbf{b}_\sigma \in \mathbb{R}^{1 \times k}$ 指的是偏置向量。

由上述可得, \mathbf{z}_u^A 是变分分布 $q_\phi(\mathbf{z}_u^A | \mathbf{x}_u^A)$ 随机采样得到的, 但在训练过程中, 这一采样操作并不可导, 也就是说没法求出梯度, 进而导致整个模型无法实现梯度的反向传播。为了解决这一问题, 本文使用重参数技巧来计算 \mathbf{z}_u^A [36-37], 即 $\mathbf{z}_u^A = \boldsymbol{\mu}_u^A + \boldsymbol{\varepsilon} \odot \boldsymbol{\sigma}_u^A$, 其中 $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \text{diag}(1))$ 。另外, 根据变分推断, 所有的变分分布都应该和标准的正态分布进行对齐, 才能保证模型更好的生成能力。为此, 变分自编码器增加了一个约束, 即使用 Kullback-Leibler 散度(KL 散度)来计算并最小化 $q_\phi(\mathbf{z}_u^A | \mathbf{x}_u^A)$ 和标准的正态分布 $p(\mathbf{z}_u)$ 之间的差距, 即

$$\mathcal{L}_{KL}(\mathbf{z}_u^A) = \text{KL}(q_\phi(\mathbf{z}_u^A | \mathbf{x}_u^A) \| p(\mathbf{z}_u)) \quad (4)$$

在获得用户潜在特征 \mathbf{z}_u^A 之后, 解码器根据 \mathbf{z}_u^A 来生成用户和所有物品的交互向量, 记为 $\hat{\mathbf{x}}_u^A$, 其计算公式如(5)所示。为了使用户的潜在特征学得更准确, $\hat{\mathbf{x}}_u^A$ 与 \mathbf{x}_u^A 应尽可能地接近, 即通过最小化重构损失函数 $\mathcal{L}(\hat{\mathbf{x}}_u^A, \mathbf{x}_u^A)$,

$$\hat{\mathbf{x}}_u^A \sim p_\theta(\mathbf{x}_u^A | \mathbf{z}_u^A) \quad (5)$$

$$\mathcal{L}(\hat{\mathbf{x}}_u^A, \mathbf{x}_u^A) = \mathbb{E}_{q_\phi(\mathbf{z}_u^A | \mathbf{x}_u^A)} \left[\log p_\theta(\mathbf{x}_u^A | \mathbf{z}_u^A) \right] \quad (6)$$

根据以上描述, 可以得到子模型 M^A 总的损失

函数, 即

$$\mathcal{L}_{M^A} = \mathcal{L}(\hat{\mathbf{x}}_u^A, \mathbf{x}_u^A) + \beta \mathcal{L}_{KL}(\mathbf{z}_u^A) \quad (7)$$

其中, β 是超参数。

4.3 无偏数据重构

在 Tri-VAE 中, 子模型 M^T 的作用是重构无偏数据 S^T 。该子模型的模型结构和 M^A 一样, 具有相同的编码器和解码器。在子模型 M^T 中, 使用 $\mathbf{z}_u^T \in \mathbb{R}^{1 \times k}$ 表示用户的潜在特征, $\boldsymbol{\mu}_u^T \in \mathbb{R}^{1 \times k}$ 和 $\boldsymbol{\sigma}_u^T \in \mathbb{R}^{1 \times k}$ 分别表示用户潜在特征的变分分布的均值和标准差。在给定 \mathbf{X}^T 的情况下, 可以得出每个用户和所有物品产生的交互向量, 记为 $\mathbf{x}_u^T \in \{0,1\}^{1 \times m}$ 。具体而言, 将 \mathbf{x}_u^T 作为 M^T 的输入, 通过多层感知器可以得到用户的潜在特征的变分分布 $q_\phi(\mathbf{z}_u^T | \mathbf{x}_u^T)$, 并由该分布随机采样得到 \mathbf{z}_u^T , 即

$$q_\phi(\mathbf{z}_u^T | \mathbf{x}_u^T) = \mathcal{N}(\boldsymbol{\mu}_u^T, \text{diag}((\boldsymbol{\sigma}_u^T)^2)) \quad (8)$$

$$\boldsymbol{\mu}_u^T = \text{ReLU}(\mathbf{x}_u^T \mathbf{w}_\mu + \mathbf{b}_\mu) \quad (9)$$

$$\boldsymbol{\sigma}_u^T = \exp(\mathbf{x}_u^T \mathbf{w}_\sigma + \mathbf{b}_\sigma) \quad (10)$$

同样地, 根据变分推断, 所有的变分分布都应该和标准的正态分布进行对齐。因此, 子模型 M^T 增加了一项约束, 即使用 KL 散度来约束 $q_\phi(\mathbf{z}_u^T | \mathbf{x}_u^T)$ 和标准的正态分布 $p(\mathbf{z}_u)$ 之间的差距,

$$\mathcal{L}_{KL}(\mathbf{z}_u^T) = \text{KL}(q_\phi(\mathbf{z}_u^T | \mathbf{x}_u^T) \| p(\mathbf{z}_u)) \quad (11)$$

另外, \mathbf{z}_u^T 由变分分布 $q_\phi(\mathbf{z}_u^T | \mathbf{x}_u^T)$ 随机采样得到, 而在训练过程中, 采样过程无法求出梯度, 从而导致无法更新模型, 故本文再次使用重参数技巧来计算 \mathbf{z}_u^T [36-37], 即 $\mathbf{z}_u^T = \boldsymbol{\mu}_u^T + \boldsymbol{\varepsilon} \odot \boldsymbol{\sigma}_u^T$ 。

在获得用户潜在特征 \mathbf{z}_u^T 后, 可以根据 \mathbf{z}_u^T 来生成用户和所有物品产生的交互向量, 记为 $\hat{\mathbf{x}}_u^T$, 其计算公式如(12)所示。接着通过最小化重构损失函数 $\mathcal{L}(\hat{\mathbf{x}}_u^T, \mathbf{x}_u^T)$ 来约束 $\hat{\mathbf{x}}_u^T$ 与 \mathbf{x}_u^T , 使它们尽可能地接近, 从而提高模型的生成能力, 即

$$\hat{\mathbf{x}}_u^T \sim p_\theta(\mathbf{x}_u^T | \mathbf{z}_u^T) \quad (12)$$

$$\mathcal{L}(\hat{\mathbf{x}}_u^T, \mathbf{x}_u^T) = \mathbb{E}_{q_\phi(\mathbf{z}_u^T | \mathbf{x}_u^T)} \left[\log p_\theta(\mathbf{x}_u^T | \mathbf{z}_u^T) \right] \quad (13)$$

根据以上描述, 可以得到子模型 M^T 总的损失函数, 即

$$\mathcal{L}_{M^T} = \mathcal{L}(\hat{\mathbf{x}}_u^T, \mathbf{x}_u^T) + \beta \mathcal{L}_{KL}(\mathbf{z}_u^T) \quad (14)$$

其中, β 是超参数。

4.4 联合数据重构

在 Tri-VAE 中, 子模型 M^C 的作用是重构联合数据 S^C , 即同时重构 S^A 和 S^T 。与 M^A 和 M^T 一样, 该子模型也是变分自编码器, 有着相同的编码器和解码器。在子模型 M^C 中, 使用 $\mathbf{z}_u^C \in \mathbb{R}^{1 \times k}$ 表示用户的潜在特征, $\boldsymbol{\mu}_u^C \in \mathbb{R}^{1 \times k}$ 和 $\boldsymbol{\sigma}_u^C \in \mathbb{R}^{1 \times k}$ 分别表示用户潜在特征的变分分布的均值和标准差。由 \mathbf{X}^C 可以得出每个用户和所有物品产生的交互向量, 记为 $\mathbf{x}_u^C \in \{0,1\}^{1 \times m}$ 。将 \mathbf{x}_u^C 作为 M^C 的输入, 通过多层感知器来得到用户潜在特征的变分分布 $q_\phi(\mathbf{z}_u^C | \mathbf{x}_u^C)$, 并由该分布随机采样得到 \mathbf{z}_u^C , 即有

$$q_\phi(\mathbf{z}_u^C | \mathbf{x}_u^C) = \mathcal{N}(\boldsymbol{\mu}_u^C, \text{diag}((\boldsymbol{\sigma}_u^C)^2)) \quad (15)$$

$$\boldsymbol{\mu}_u^C = \text{ReLU}(\mathbf{x}_u^C \mathbf{w}_\mu + \mathbf{b}_\mu) \quad (16)$$

$$\boldsymbol{\sigma}_u^C = \exp(\mathbf{x}_u^C \mathbf{w}_\sigma + \mathbf{b}_\sigma) \quad (17)$$

类似地, 根据变分推断, 本文使用 KL 散度来约束 $q_\phi(\mathbf{z}_u^C | \mathbf{x}_u^C)$ 和标准的正态分布 $p(\mathbf{z}_u)$ 之间的差距, 以满足用户潜在特征的先验分布为标准高斯分布的假设, 即

$$\mathcal{L}_{KL}(\mathbf{z}_u^C) = \text{KL}(q_\phi(\mathbf{z}_u^C | \mathbf{x}_u^C) \| p(\mathbf{z}_u)) \quad (18)$$

另外, 本文使用重参数技巧来计算 \mathbf{z}_u^C [36-37], 即 $\mathbf{z}_u^C = \boldsymbol{\mu}_u^C + \boldsymbol{\varepsilon} \odot \boldsymbol{\sigma}_u^C$, 以此来解决在训练过程中, 采样过程无法计算梯度的问题。

从信息的角度来看, 联合数据是由有偏数据和无偏数据组成的, 那么它所包含的信息是最丰富的, 故在 Tri-VAE 中, 将 M^C 作为目标子模型。然而, 因为有偏数据和无偏数据的规模不同, 有偏数据占据了大部分。因此, 为了获得更精准的无偏的个性化推荐模型, M^C 应当有效地缓解偏置的影响。然而, 偏置本身是隐蔽的, 无法直接通过某种测量方式来衡量, 所以只能从间接的角度来解决这一问题。

先前的大部分研究工作认为偏置隐式地反映在标签或模型参数上 [5,9-13], 并从标签的角度或从模型参数的角度上来缓解偏置的影响。很少有研究工作从特征的角度、潜在特征的角度、或结合多个角度来解决偏置问题。

受潜在特征的性质的启发, 不同类型的数据所对应的潜在特征不同。故本文从潜在特征的角度来缓解偏置问题, 即认为偏置实际上隐藏在用户的潜

在特征中。也就是说, 由子模型 M^A 和 M^C 得到的 $\boldsymbol{\mu}_u^A$ 和 $\boldsymbol{\mu}_u^C$ 是相对有偏的, 而子模型 M^T 得到的 $\boldsymbol{\mu}_u^T$ 是相对无偏的。因为 $\boldsymbol{\mu}_u^A$ 是从有偏数据学习得到的, 而 $\boldsymbol{\mu}_u^T$ 是从无偏数据学习得到的。对于 $\boldsymbol{\mu}_u^C$ 而言, 尽管它是从联合数据学习而来, 但联合数据中大部分都属于有偏数据。那么此时, 可以通过衡量 $\boldsymbol{\mu}_u^A$, $\boldsymbol{\mu}_u^T$ 和 $\boldsymbol{\mu}_u^C$ 的之间的差异来间接地缓解偏置的影响。因此, 基于这样的动机, 本文设计了一个特征校正模块 (Feature Calibration Module, FCM), 即以 $\boldsymbol{\mu}_u^C$ 为基准, 利用 $\boldsymbol{\mu}_u^A$ 和 $\boldsymbol{\mu}_u^T$ 之间的差异来消除 $\boldsymbol{\mu}_u^C$ 中的偏置特征, 从而获得无偏的用户潜在特征。

本文使用 $\hat{\boldsymbol{\mu}}_u^C$ 来表示无偏的用户潜在特征, 并假设 $\hat{\boldsymbol{\mu}}_u^C$ 和 $\boldsymbol{\mu}_u^C$ 之间的差异是线性的, 即

$$\hat{\boldsymbol{\mu}}_u^C = \boldsymbol{\mu}_u^C - \mathbf{d} \quad (19)$$

其中, $\mathbf{d} \in \mathbb{R}^{1 \times k}$ 指的是 $\hat{\boldsymbol{\mu}}_u^C$ 和 $\boldsymbol{\mu}_u^C$ 的差异。

另外, 因为 $\boldsymbol{\mu}_u^A$ 是相对有偏的, 而 $\boldsymbol{\mu}_u^T$ 是相对无偏的, 所以可以通过衡量它们之间的差异来获得偏置的信息。换句话说, $\hat{\boldsymbol{\mu}}_u^C$ 和 $\boldsymbol{\mu}_u^C$ 之间的差异是一个关于 $\boldsymbol{\mu}_u^A$ 和 $\boldsymbol{\mu}_u^T$ 的函数, 即

$$\mathbf{d} = f(\boldsymbol{\mu}_u^A, \boldsymbol{\mu}_u^T) \quad (20)$$

其中, $f(\boldsymbol{\mu}_u^A, \boldsymbol{\mu}_u^T)$ 指的是关于 $\boldsymbol{\mu}_u^A$ 和 $\boldsymbol{\mu}_u^T$ 的任意函数。

根据以上描述, 如果知道了 $f(\boldsymbol{\mu}_u^A, \boldsymbol{\mu}_u^T)$ 的具体形式, 就可以获得无偏的用户潜在特征。但是从偏置的隐蔽性可以得出, $f(\boldsymbol{\mu}_u^A, \boldsymbol{\mu}_u^T)$ 是一个非常复杂的函数, 其具体形式是无法得到的。不过, $f(\boldsymbol{\mu}_u^A, \boldsymbol{\mu}_u^T)$ 可以通过神经网络来拟合, 因为神经网络对函数具有较强的拟合能力。因此, 本文基于神经网络设计了特征校正模块。具体来说, 首先将 $\boldsymbol{\mu}_u^A$ 和 $\boldsymbol{\mu}_u^T$ 进行拼接, 再通过具有激活函数的单层全连接网络来提取具有偏置信息的特征, 最后根据公式(19), 和 $\boldsymbol{\mu}_u^C$ 作差, 进而获得无偏的用户潜在特征 $\hat{\boldsymbol{\mu}}_u^C$, 即

$$f(\boldsymbol{\mu}_u^A, \boldsymbol{\mu}_u^T) = \mathbf{g}([\boldsymbol{\mu}_u^A; \boldsymbol{\mu}_u^T] \mathbf{w}_f + \mathbf{b}_f) \quad (21)$$

$$\hat{\boldsymbol{\mu}}_u^C = \boldsymbol{\mu}_u^C - f(\boldsymbol{\mu}_u^A, \boldsymbol{\mu}_u^T) \quad (22)$$

其中, $\mathbf{g}(\cdot)$ 指的是一个具体的激活函数, $\mathbf{w}_f \in \mathbb{R}^{2k \times k}$ 和 $\mathbf{b}_f \in \mathbb{R}^{1 \times k}$ 分别是特征校正模块的权重矩阵和偏置向量。

因此, 由 $\hat{\mu}_u^C$ 和 μ_u^C 可以得到无偏的用户潜在特征的变分分布 $q_\phi(\hat{z}_u^C | \mathbf{x}_u^C)$, 并从该分布随机采样可以得到无偏的用户潜在特征 \hat{z}_u^C , 如公式(23)所示。

$$q_\phi(\hat{z}_u^C | \mathbf{x}_u^C) = \mathcal{N}(\hat{\mu}_u^C, \text{diag}((\sigma_u^C)^2)) \quad (23)$$

此外, 对于 \hat{z}_u^C 的计算, 它是由变分分布采样得到的, 但在训练过程中, 采样过程存在无法计算梯度的问题。因此, 本文同样地使用重参数技巧来计算 \hat{z}_u^C [36-37], 即 $\hat{z}_u^C = \hat{\mu}_u^C + \varepsilon \odot \sigma_u^C$ 。

在子模型 M^C 中, 本文使用 $\hat{\mathbf{x}}_u^C$ 表示由解码器生成的用户和所有物品产生的交互向量, 其计算公式如(24)所示。之后再通过最小化重构损失函数来优化模型。但不同于子模型 M^A 和 M^T 直接对输入的数据进行重构, 子模型 M^C 在重构过程中, 并没有直接让 $\hat{\mathbf{x}}_u^C$ 和 \mathbf{x}_u^C 尽可能地接近, 而是对有偏数据 S^A 和无偏数据 S^T 作了区别对待。理想情况下, M^C 的输出应当更接近无偏, 而 S^A 本身是有偏的。受偏置信息反映在标签上的启发, 如果直接对 S^A 进行重构可能会影响模型最后的推荐性能。因此, 借鉴文献[11]中的 Weight 策略的思路, 本文在重构 S^A 时设置置信度来控制, 以保证模型的无偏性。具体而言, 本文为重构 \mathbf{x}_u^A 的损失函数设置了权重 $\alpha \in (0, 1)$, 而将重构 \mathbf{x}_u^T 的损失函数的权重设置为 1, 即有

$$\hat{\mathbf{x}}_u^C \sim p_\theta(\mathbf{x}_u^C | \mathbf{z}_u^C) \quad (24)$$

$$\alpha \mathcal{L}(\hat{\mathbf{x}}_u^C, \mathbf{x}_u^A) = \alpha \mathbb{E}_{q_\phi(\hat{z}_u^C | \mathbf{x}_u^C)} \left[\log p_\theta(\mathbf{x}_u^A | \hat{z}_u^C) \right] \quad (25)$$

$$\mathcal{L}(\hat{\mathbf{x}}_u^C, \mathbf{x}_u^T) = \mathbb{E}_{q_\phi(\hat{z}_u^C | \mathbf{x}_u^C)} \left[\log p_\theta(\mathbf{x}_u^T | \hat{z}_u^C) \right] \quad (26)$$

另外, 在本文提出的 Tri-VAE 中, 由于 M^T 是在重构无偏数据, 故 M^T 的预测输出可以被认为是相对无偏的标签。受文献[11]中的启发, M^T 的预测输出包含了较为丰富的无偏信息。因此, 为了加强对无偏数据中的信息的利用, 借鉴于文献[11]中的 Bridge

算法 1. 三任务变分自编码器的训练过程

输入: 用户与物品的交互矩阵 \mathbf{X}^A 、 \mathbf{X}^T 和 \mathbf{X}^C , 用户总数 n , 物品总数 m , 潜在特征维度 $rank$, 正则化项权重 λ , 重构损失函数权重 α 和 γ , KL 约束项权重 β , dropout rate ρ , 最大迭代次数 *iteration number*, 学习率 *learning rate*, 一次训练所选取的样本数 *batch size*

输出: 编码器 ϕ , 解码器 θ , 特征校正模块 FCM

1. 随机初始化编码器 ϕ , 解码器 θ 和特征校正模块 FCM
2. FOR *iter* = 1: *iteration number* DO
3. 从 \mathcal{D} 中随机采样和 S^A 规模大小一样的(用户, 物品)对, 并对用户 ID 和物品 ID 进行标记
4. 根据 n 和 *batch size*, 按照用户 ID 顺序分成多个 *batches*
5. FOR *batch* = 1: *length(batches)* DO
6. FOR 对于 *batch* 中的所有用户 DO
7. 分别从 \mathbf{X}^A 、 \mathbf{X}^T 和 \mathbf{X}^C 取出用户 u 对应的交互向量 \mathbf{x}_u^A 、 \mathbf{x}_u^T 和 \mathbf{x}_u^C
8. 根据 dropout 操作对 \mathbf{x}_u^A 、 \mathbf{x}_u^T 和 \mathbf{x}_u^C 进行处理
9. 根据公式(2)和(3)求出 μ_u^A 和 σ_u^A
10. 根据公式(9)和(10)求出 μ_u^T 和 σ_u^T
11. 根据公式(16)和(17)求出 μ_u^C 和 σ_u^C
12. 将 μ_u^A 和 μ_u^T 进行拼接, 根据公式(21)和(22)求出 $\hat{\mu}_u^C$
13. 从标准高斯分布随机采样 ε , 根据重参数技巧求出 \mathbf{z}_u^A 、 \mathbf{z}_u^T 和 $\hat{\mathbf{z}}_u^C$
14. 根据公式(5)、(12)和(24)重构输入数据, 即 $\hat{\mathbf{x}}_u^A$ 、 $\hat{\mathbf{x}}_u^T$ 和 $\hat{\mathbf{x}}_u^C$
15. 根据 $\hat{\mathbf{x}}_u^T$ 生成用户与物品的伪交互向量 $\tilde{\mathbf{x}}_u^T$
16. 根据公式(29)求出总的损失函数 $\mathcal{L}_{\text{Tri-VAE}}$, 进而求出梯度
17. END FOR
18. 对 *batch* 中所求出的所有梯度进行平均
19. 对编码器 ϕ , 解码器 θ 和特征校正模块 FCM 进行更新
20. END FOR
21. END FOR
22. RETURN 编码器 ϕ , 解码器 θ 和特征校正模块 FCM

策略的思路, 本文设计了一个标签增强模块(Label Enhancement Module, LEM), 旨在训练过程中加强利用 M^T 的预测输出。

本文使用 \mathcal{D} 表示所有用户和物品交互的集合, 包括可观测到的和不可观测到的(用户, 物品)对。在训练过程中, 每一次迭代都从 \mathcal{D} 中随机采样和 S^A 规模大小一样的(用户, 物品)对, 再利用 M^T 对这些(用户, 物品)对进行预测得到相应的伪标签, 即用户

对物品的反馈。本文将这一个伪数据集记为 \tilde{S}^T 。另外, 本文使用 $\tilde{\mathbf{x}}_u^T \in \{0,1\}^{l \times m}$ 表示由 \tilde{S}^T 得到的每个用户和所有物品产生的交互向量。本文期望 M^C 的预测输出 $\hat{\mathbf{x}}_u^C$ 和 $\tilde{\mathbf{x}}_u^T$ 尽可能地接近, 为此构造了它们之间的重构损失函数 $\mathcal{L}(\hat{\mathbf{x}}_u^C, \tilde{\mathbf{x}}_u^T)$, 如公式(27)所示。类似地, 本文在该重构损失函数设置了权重 $\gamma \in (0,1)$, 因为无偏数据规模通常较小, 不足以让无偏模型 M^T 得到充分的训练, 所以 \tilde{S}^T 中的伪标签不一定是正确的。因此, 为了解决这一问题, 本文设置了权重 γ 加以调整。

$$\gamma \mathcal{L}(\hat{\mathbf{x}}_u^C, \tilde{\mathbf{x}}_u^T) = \gamma \mathbb{E}_{q_\theta(\mathbf{z}_u^C | \mathbf{x}_u^C)} \left[\log p_\theta(\tilde{\mathbf{x}}_u^T | \mathbf{z}_u^C) \right] \quad (27)$$

根据以上描述, 可以得到子模型 M^C 总的损失函数, 即

$$\mathcal{L}_{M^C} = \alpha \mathcal{L}(\hat{\mathbf{x}}_u^C, \mathbf{x}_u^A) + \mathcal{L}(\hat{\mathbf{x}}_u^C, \mathbf{x}_u^T) + \beta \mathcal{L}_{KL}(\mathbf{z}_u^T) + \gamma \mathcal{L}(\hat{\mathbf{x}}_u^C, \tilde{\mathbf{x}}_u^T) \quad (28)$$

其中, α , β 和 γ 都是超参数。

4.5 优化

结合以上所有描述, 本文采用联合训练的方式来训练模型, 并采用批量梯度下降(BGD)来最小化各个子模型的损失函数之和, 即

$$\mathcal{L}_{\text{Tri-VAE}} = \mathcal{L}_{M^A} + \mathcal{L}_{M^T} + \mathcal{L}_{M^C} \quad (29)$$

另外, 整个模型的训练过程如算法 1 所示。

5 实验结果与分析

5.1 数据集

本文对解决偏置问题的有效性进行评估, 故所采用的数据集必须包含无偏数据。目前已公开的数据集只有两个, 即 Yahoo! R3^[38]和 Coat Shopping^[9], 关于它们的具体信息如表 1 所示。

- Yahoo! R3^[38]: 这是一个关于用户对音乐评分的数据集, 包含了一个有偏的常规数据子集和一个无偏的随机数据子集。其中, 常规数据子集包含 15400 个用户, 1000 首歌曲和大约 30 万条评分记录, 评分范围是 $\{1,2,3,4,5\}$ 。常规数据子集是由系统使用常规的推荐策略进行推荐, 用户根据自身选择来对歌曲进行评价而收集到的, 故被认为是有偏的。随机数据子集是通过要求前 5400 个用户对 10 首随机选择的歌曲进行评分收集而来的, 故被认为是无偏的。因为本文研究的问题是一元反馈的偏置问题, 故需

将评分转换成一元反馈。因此, 本文对数据集进行了预处理。如果评分大于 3 分, 那么该记录被认为是正反馈^[10-11], 即 $r_{ui} = 1$ 。在实验中, 本文将常规数据子集作为有偏数据(S^A), 并将随机数据子集随机划分成 3 个子集, 其中 10% 作为目标训练的无偏数据(S^T), 10% 作为验证集用以选择超参数, 最后的 80% 作为测试集用以评估模型的性能。

- Coat Shopping^[9]: 这是一个关于用户对外衣(coat)评分的数据集, 是通过让 Amazon Mechanical Turk 平台上的 290 用户对 300 件外衣进行评分收集而来的。和 Yahoo! R3 一样, 该数据集包含了一个有偏的常规数据子集和一个无偏的随机数据子集。其中, 常规数据子集是用户根据自身选择来对外衣进行评价而收集的, 大约有 7000 条评分记录, 评分范围是 $\{1,2,3,4,5\}$, 被认为是有偏的。随机数据子集是通过要求所有用户对 16 件随机选择的外衣进行评分收集的, 被认为是无偏的。常规数据子集和随机数据子集的预处理及划分方式和 Yahoo! R3 一样。

表 1 实验所采用的数据集的统计信息。其中 “#Feedback” 指的是反馈样本的数量, “P/N” 指的是数据集中正反馈和负反馈的比例

Table1 Statistics of the datasets used in the experiments. Notice that #Feedback denotes the number of feedback samples and P/N denotes the ratio between the numbers of positive feedback and negative feedback

	Yahoo! R3		Coat Shopping	
	#Feedback	P/N(%)	#Feedback	P/N(%)
S^A	311704	67.02	6960	37.69
S^T	5400	9.05	464	28.89
S_{va}	5400	9.31	464	23.40
S_{te}	43200	9.76	3712	21.94

5.2 评价指标

为了评估模型的推荐性能, 本文采用推荐系统中常用的评价指标进行评估, 包括 ROC(Receiver Operating Characteristic, ROC)曲线下的面积(Area Under Roc Curve, AUC)、归一化折损累积增益(Normalized Discounted Cumulative Gain, NDCG)、精确率(Precision@N, P@N)和召回率(Recall@N, R@N)

4 个指标, 各个指标的定义如下所示。其中, 在实验过程中, N 设置为 10、20 和 50。另外, 在实验过程中, 本文将 AUC 作为主要指标, 其他作为参考指标^[10-12]。

下面对 4 个指标的定义进行一一阐述。

- ROC 曲线下的面积(AUC): 其计算公式如下所示

$$AUC = \frac{\sum_{(i,j) \in \Omega}^{L_p} Rank_{ij} - \binom{L_p}{2}}{L_p(L - L_p)}$$

其中, Ω 指的是测试集上(在选择参数过程中为验证集)的所有正反馈的集合, L_p 指的是 Ω 中正反馈的总数, $Rank_{ij}$ 指的是根据预测结果进行降序排列得到的正反馈 (i, j) 在所有的反馈中的排序位置, L 指的是在测试集上(在选择参数过程中为验证集)的反馈总数, 包括正反馈和负反馈。

- 归一化折损累积增益(NDCG): 其计算公式如下

$$NDCG = \frac{\sum_{u \in \mathcal{U}^e} NDCG_u @K}{|\mathcal{U}^e|}$$

$$NDCG_u @K = \frac{DCG_u @K}{IDCG_u}$$

$$DCG_u @K = \sum_{i=1}^K \frac{rel_i}{\log_2(i+1)}$$

其中, \mathcal{U}^e 指的是测试集上(在选择参数过程中为验证集)的用户集合, $NDCG_u @K$ 指的是用户 u 的归一化折损累积增益, $DCG_u @K$ 指的是用户 u 的折损累积增益, $IDCG_u$ 指的是理想情况下用户 u 的折损累积增益, 即根据返回的结果, 按照真实的相关性重新排序, 进而计算 $DCG_u @K$ 得来的, K 指的是推荐列表的长度, rel_i 指的是推荐物品的相关性。在实验过程中, K 设置为 50。

- 精确率(P@N): 其计算公式如下

$$P@N = \frac{\sum_{u \in \mathcal{U}} |R(u) \cap T(u)|}{\sum_{u \in \mathcal{U}} |R(u)|}$$

其中, $R(u)$ 指的是模型给用户生成的长度为 N 的推荐列表, $T(u)$ 指的是用户在测试集上(在选择参数过程中为验证集)的反馈形成的推荐列表。

- 召回率(R@N): 其计算公式如下

$$R@N = \frac{\sum_{u \in \mathcal{U}} |R(u) \cap T(u)|}{\sum_{u \in \mathcal{U}} |T(u)|}$$

其中, $R(u)$ 指的是模型给用户生成的长度为 N 的推

荐列表, $T(u)$ 指的是用户在测试集上(在选择参数过程中为验证集)的反馈形成的推荐列表。

5.3 基线模型

本文提出的模型是在 VAE 的基础上进行改进的, 故将 VAE 作为最相关的基线模型。不同类型的数据进行训练会产生不同版本的 VAE。在本文中, 有以下 3 个版本, 即分别使用有偏数据 S^A 、无偏数据 S^T 和联合数据 S^C 进行训练, 本文将它们依次记为 VAE(S^A)、VAE(S^T)和 VAE(S^C)。除此之外, 本文还将近年来提出的具有代表性的方法作为基线模型, 分别是基于逆倾向性得分的模型 (IPS)^[9,28-29]、CausE^[10]和 KDCRec^[11]。

- 基于逆倾向性得分的模型(IPS)^[9,28-29]: 基于逆倾向性得分消除偏置的算法是目前学术界比较主流的方法, 通过对训练集中的每个样本赋予一个倾向性得分, 从而来训练一个无偏的模型。倾向性得分的计算方法有很多^[13,28-29], 本文采用的是其中一个具有代表性的方法, 即朴素贝叶斯评估方法。该方法通过朴素贝叶斯来计算物品的曝光概率, 以此得到每个样本的倾向性得分, 具体计算公式如下:

$$P_i = \frac{P(\mathcal{I} = i | O = 1)P(O = 1)}{P(\mathcal{I} = i)}$$

其中, $P(\mathcal{I} = i | O = 1)$ 指的是在可观测到的样本中, 所有用户对物品 i 产生交互的样本所占的比例; $P(O = 1)$ 指的是在全部样本中(包括未观测样本), 可观测的样本所占的比例; $P(\mathcal{I} = i)$ 指的是在无偏数据中, 所有用户对物品 i 产生交互的样本所占的比例。另外, 为了实验的公平性, 本文使用 VAE 作为基本模型。

- CausE^[10]: 其思想是通过正则化项来约束有偏模型和无偏模型的参数, 进而缓解偏置的影响。首先通过无偏数据进行训练得到无偏模型, 接着使用有偏数据训练新的模型, 同时以无偏模型的模型参数作为参考, 使用正则化项进行约束。同样地, 为了公平比较, 本文使用 VAE 作为 CasuE 的基本模型。
- KDCRec^[11]: 一个基于知识蒸馏的框架。作者在文中提出了多种策略, 其中最具有代表性的策略是 Bridge 策略。因为从作者提供的实验结果来看, 使用 Bridge 策略训练出来的模型的推荐性能表现最佳。因此, 本文采用 Bridge 策略作为基线模型之一。类似地, 为了公平比较,

本文使用 VAE 作为基本模型。

5.4 实验设置

本文使用 Python 3.6 和 TensorFlow 1.14 进行实验, 模型训练和测试都是在搭载了 Ubuntu18.04 系统的计算机集群上进行的。该集群包含 2 个 26 核的 CPU, 256GB 内存和 2 个 Nvidia Tesla P100 GPU。在 Tri-VAE 中, 对于特征校正模块中的激活函数 $g(\cdot)$, 本文采用了 \tanh 函数和 sigmoid 函数。本文使用网格搜索的方式对超参数进行搜索, 并通过验证集上 AUC 的值来确定最佳的超参数。其中, 超参数的设置范围如表 2 所示。另外, 对于 Coat Shopping 数据

集, 由于验证集的规模较小, 为了让实验结果更具有说服力, 在选参的过程中, 先使用网格搜索的方式进行搜索, 再选择在验证集上 AUC 性能最好的前 5 组参数, 让每组参数运行 5 遍, 最后选择平均性能最好的那组参数作为最佳的超参数。此外, 在模型评估过程中, 本文通过改变随机种子, 对每个模型分别进行 10 次训练和测试, 最后将平均值作为最终的测试结果。同时, 本文还使用双样本 t 检验对各个指标进行了显著性检验, 即在各个指标上检验比较本文提出的模型的结果和最好的基线模型的结果。本文所涉及的实验数据、源代码和脚本将会公开, 可从 <http://csse.szu.edu.cn/staff/panwk/Tri-VAE/> 获得。

表 2 实验过程中超参数的设置范围。

Table 2 The ranges of the values for the hyper-parameters to be tuned in the experiments.

参数名	范围		用途
	Yahoo! R3	Coat Shopping	
$rank$	{50,100,200}		潜在特征维度
λ	{0.00001,0.0001,0.001,0.01,0.1}		正则化项上的权重
α	{0.1,0.2,0.3,0.4,0.5,0.6}		$\mathcal{L}(\hat{x}_u^C, x_u^A)$ 上的权重
β	{0.2}		$\mathcal{L}_{KL}(\cdot)$ 上的权重
γ	{0.0001,0.001,0.01,0.1}		$\mathcal{L}(\hat{x}_u^C, \tilde{x}_u^T)$ 上的权重
ρ	{0.2,0.3,0.4,0.5}		dropout rate
iteration number	300		迭代次数
learning rate	0.0001		学习率
batch size	100	{5,10,15,20}	一次训练所选取的样本数

5.5 实验结果分析

5.5.1 性能比较

对于数据集 Yahoo! R3 和 Coat Shopping, 本文提出的模型 Tri-VAE 和基线模型的推荐性能比较如表 3 和表 4 所示。其中, Tri-VAE-tanh 表示特征校正模块使用 \tanh 函数作为激活函数, Tri-VAE-sig 表示特征校正模块使用 sigmoid 函数作为激活函数。从实验结果可以看出: (1)不管是 Yahoo! R3 还是 Coat Shopping, 在绝大多数情况下, Tri-VAE 的推荐性能比所有基线模型都要好。注意 AUC 是最关键的指标, 且模型参数是通过 AUC 进行选择。另外, 本文通过双样本 t 检验进行了显著性检验, 发现大部分指标的 p 值都小于 0.01。这说明本文所提出的模型能够更好地对有偏数据和无偏数据进行联合建模, 有效地利用无偏数据中的信息来缓解偏置带来的影响; (2)对于本文提出的特征校正模块, 使用不同的激活函数对最终的推荐性能有一定的影响。从主要指标来看, 在 Yahoo! R3 上, \tanh 函数的推荐性能更好, 而在 Coat

Shopping 上, 则是 sigmoid 函数的性能更好。造成这种差异的原因可能是不同的激活函数所提取到的偏置信息有所不同, 而这两个数据集包含的偏置信息存在一定的差异; (3)关于使用不同类型的数据进行训练产生的 3 个版本的 VAE, 其实验效果是 $\text{VAE}(S^C) > \text{VAE}(S^A) > \text{VAE}(S^T)$, 这说明有偏数据和无偏数据之间具有互补性。而本文提出的 Tri-VAE 比这三个基准模型的效果都要好, 说明 Tri-VAE 能更好地对有偏数据和无偏数据之间的互补性进行建模; (4)IPS-VAE 的实验效果比较差, 这说明本文中所采用的利用无偏数据和朴素贝叶斯法来计算倾向性得分并不能很好地解决偏置问题; (5)CausE 和 KDCRec 的实验效果比 $\text{VAE}(S^A)$ 要好, 但比 $\text{VAE}(S^C)$ 差, 这是因为这两个基线模型都需要利用单独使用无偏数据得到的预训练模型, 而由于无偏数据的规模较小, 导致预训练模型不能得到充分的训练。

5.5.2 消融实验

为了进一步研究 Tri-VAE, 本文设计了消融实

验。在本文所提出的模型中,子模型之间通过特征校正模块和标签增强模块进行关联。简而言之,通过利用 μ_u^T 、 μ_u^A 和 \tilde{x}_u^T 的信息来协助目标子模型 M^C 的训练。因此,本文首先一一去除 μ_u^T 、 μ_u^A 和 \tilde{x}_u^T 中的一个,分别记为“-T”、“-A”和“-P”,接着再去除它们之间的组合,包括同时去除 μ_u^T 和 μ_u^A (记为“-TA”),以及同时去除 μ_u^T 、 μ_u^A 和 \tilde{x}_u^T (记为“-TAP”)。从表 3 和表 4 的分析来看,在 Yahoo! R3 上采用 tanh 函数的推荐性能更好,而在 Coat Shopping 上则是采用 sigmoid 函数。因此,为了实验的公平性,在 Yahoo! R3 上的消融实验选择 tanh 函数作为激活函数,在 Coat Shopping 上的消融实验则选择 sigmoid 函数作为激活函数,具体的实验结果如图 2 所示。从实验结果可以看出,去除 Tri-VAE 中的某一部分或者同时去

除某几部分,都会使推荐性能变差。其中,去除 \tilde{x}_u^T 之后,模型的推荐性能小幅度下降,而去除 μ_u^T 或者 μ_u^A 之后,模型的推荐性能大幅度下降。这说明特征校正模块在 Tri-VAE 中起到了重要的作用,而标签增强模块是辅助作用,同时也验证了从潜在特征的角度来缓解偏置的有效性。另外,对于特征校正模块和标签增强模块对推荐性能的作用程度不同,这是因为特征校正模块是直接对用户的潜在特征进行校正,而标签增强模块通过标签增强的方式间接地对用户的潜在特征进行校正。

5.5.3 超参数敏感性分析

本文所提出的模型引入了两个主要的超参数,分别是 α 和 γ , 因此本文设计了以下实验对超参数进行分析。首先,为了分析 α , 先固定其他超参数,

表 3 在数据集 Yahoo! R3 上,本文提出的模型和基线模型的推荐性能比较,其中最佳结果用粗体标记,次优结果用下划线标记。另外,显著性检验中 p 值小于 0.01 的最佳结果用星号(*)标记

Table 3 Recommendation performance of our proposed model and the baselines on Yahoo! R3, where the best results are marked in bold and the second best results are marked underlined. In addition, the significantly best results are marked with a star (*), where the p -value is smaller than 0.01

方法	AUC	NDCG	P@10	P@20	P@50	R@10	R@20	R@50
VAE(S^A)	0.7693	0.0994	0.0159	<u>0.0120</u>	0.0081	0.1097	0.1618	0.2661
VAE(S^T)	0.5688	0.0261	0.0033	0.0035	0.0029	0.0199	0.0435	0.0856
VAE(S^C)	0.7725	<u>0.1000</u>	0.0159	<u>0.0120</u>	0.0081	<u>0.1094</u>	<u>0.1622</u>	0.2673
IPS-VAE	0.7457	0.0824	0.0114	0.0097	0.0073	0.0793	0.1300	0.2387
CausE	0.7706	0.0977	0.0150	0.0116	<u>0.0082</u>	0.1024	0.1576	0.2690
KDCRec	0.7731	0.0969	0.0148	0.0115	<u>0.0082</u>	0.1006	0.1569	0.2677
Tri-VAE-tanh	0.7823*	0.1025*	<u>0.0157</u>	0.0121	0.0083*	0.1090	0.1646	0.2742*
Tri-VAE-sig	<u>0.7759</u>	0.0991	0.0156	0.0119	<u>0.0082</u>	0.1061	0.1605	<u>0.2692</u>

表 4 在数据集 Coat Shopping 上,本文提出的模型和基线模型的推荐性能比较,其中最佳结果用粗体标记,次优结果用下划线标记。另外,显著性检验中 p 值小于 0.01 的最佳结果用星号(*)标记

Table 4 Recommendation performance of our proposed model and the baselines on Coat Shopping, where the best results are marked in bold and the second best results are marked underlined. In addition, the significantly best results are marked with a star (*), where the p -value is smaller than 0.01

方法	AUC	NDCG	P@10	P@20	P@50	R@10	R@20	R@50
VAE(S^A)	0.6179	0.0931	0.0186	0.0170	0.0154	0.0636	0.1175	0.2434
VAE(S^T)	0.5402	0.0769	0.0149	0.0162	0.0124	0.0460	0.1014	0.1933
VAE(S^C)	0.6253	0.0950	0.0201	0.0173	0.0158	0.0687	0.1187	0.2477
IPS-VAE	0.5747	0.0856	0.0208	0.0163	0.0140	0.0622	0.0977	0.2172
CausE	0.6207	0.0955	0.0185	0.0176	0.0159	0.0648	0.1204	0.2512
KDCRec	0.6207	0.0956	0.0185	0.0176	0.0159	0.0648	0.1206	0.2512
Tri-VAE-tanh	<u>0.6676</u>	<u>0.1177</u>	0.0242	0.0214	0.0180*	<u>0.0915</u>	<u>0.1673</u>	0.3345*
Tri-VAE-sig	0.7359*	0.1253*	0.0284*	0.0225*	<u>0.0173</u>	0.1077*	0.1756*	<u>0.3249</u>

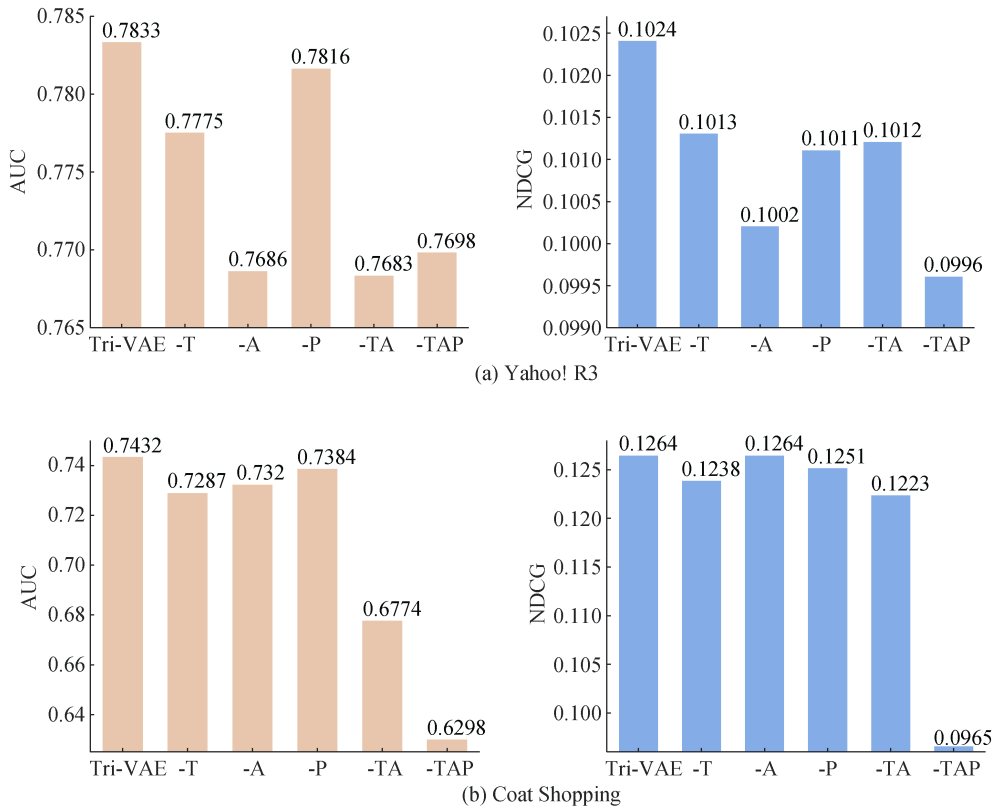
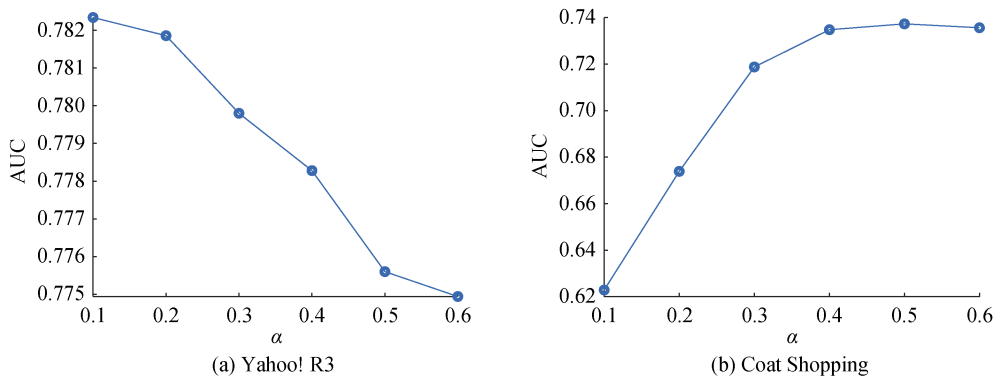


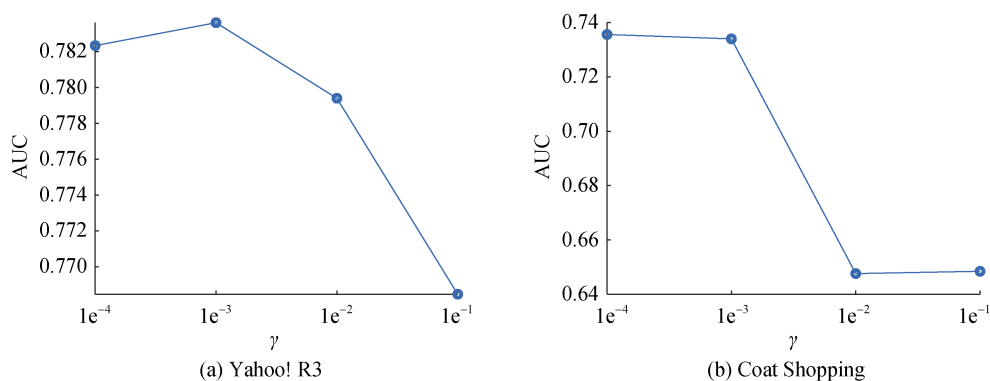
图 2 消融实验中推荐性能的比较

Figure 2 Recommendation performance in ablation studies

α 值依次从 0.1 变化到 0.6, 重新训练并测试模型, 最后的实验结果如图 3 所示。从图 3 可以看出, 最优的 α 值在数据集 Yahoo! R3 和 Coat Shopping 上分别是 0.1 和 0.5。 α 值的大小可能和有偏数据与无偏数据的数据规模之比(即 $|S^A|/|S^T|$) 有关系。因为有偏数据与无偏数据规模差异直接影响到对 S^A 的重构损失函数与对 S^T 的重构损失函数的差异。当有偏数据规模较大时, 对 S^A 的重构损失函数的影响相对会大一些, 此时为了削弱 S^A 的重构损失函数的影响, α

值就应当相对小一些。接着分析 γ , 类似地, 先固定其他超参数, γ 值依次从 0.0001 变化到 0.1, 重新训练并测试模型, 最后的实验结果如图 4 所示。从图 4 可以看出, 最优的 γ 值在数据集 Yahoo! R3 和 Coat Shopping 上分别是 0.001 和 0.0001, γ 值越大, 最终的推荐性能越差。这和无偏数据的数据规模有关。尽管通过无偏数据可以训练得到相对比较无偏的模型, 但由于无偏数据的规模较小, 较难充分训练得到一个很准确的预测结果, 也就是说, 标签增强模块产生的伪标签的可信度不会很高, 故 γ 值不会很大。

图 3 不同 α 值对推荐性能的影响Figure 3 Recommendation performance of our proposed model with different values of α

图 4 不同 γ 值对推荐性能的影响Figure 4 Recommendation performance of our proposed model with different values of γ

5.5.4 模型的收敛性分析

为了进一步分析本文提出的模型 Tri-VAE 的收敛性, 本文进行了实验验证。通过 5.5.1 节的实验结果可知, 在众多的基线模型中, $\text{VAE}(S^c)$ 的模型性能最好, 同时 VAE 也是与 Tri-VAE 最相关的基线模型。因此, 本文通过分析 Tri-VAE 和 $\text{VAE}(S^c)$ 这两个模型在训练时总的损失和验证集上的 AUC 随迭代次数的变化情况来验证模型的收敛性。具体实验结果如

图 5 所示。从实验结果可以看出, 本文提出的模型的收敛性较好, 收敛速度不会随着模型复杂度的增加而减缓。首先, 在 Yahoo! R3 数据集上, Tri-VAE 和 $\text{VAE}(S^c)$ 在训练时总的损失的收敛趋势较为相近。而对于验证集上的 AUC, Tri-VAE 的收敛性要优于 $\text{VAE}(S^c)$ 。其次, 在 Coat Shopping 数据集上, 与 Yahoo! R3 数据集上的结果类似, Tri-VAE 和 $\text{VAE}(S^c)$ 在训练时总的损失的收敛性相当。而在验证集

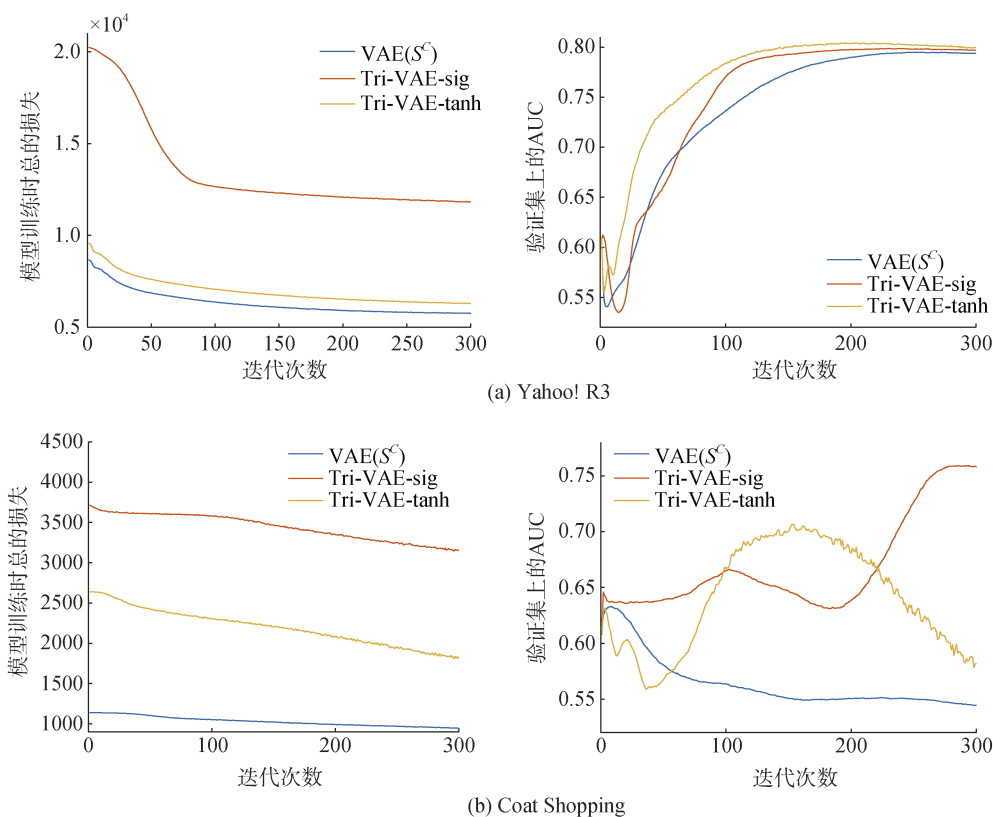


图 5 模型训练时总的损失和验证集上 AUC 随迭代次数的变化情况

Figure 5 Variation of the total loss when training the model and the AUC on the validation set with the number of iterations

上的 AUC, 却表现出了不一样的结果。通过比较发现, $VAE(S^c)$ 出现了过拟合的现象, 即在训练集上总的损失越来越小, 但验证集上的 AUC 却在变差。而本文提出的模型(Tri-VAE-sig)不仅能够正常收敛, 而且没有出现过拟合的现象。

5.5.5 关于特征校正模块的进一步探讨

由 5.5.1 节和 5.5.2 节可以得出, 在本文提出的模型中, 特征校正模块起到了关键的作用。因此, 为了进一步探索特征校正模块, 本文通过改变特征校正模块的结构来进行实验分析。本文首先改变函数 $f(\mu_u^A, \mu_u^T)$ 的具体形式。由 4.4 节可得, 特征校正模块的动机是通过衡量 μ_u^A 和 μ_u^T 之间的差异来消除 μ_u^C 中的偏置特征, 那么直觉的做法就是先让 μ_u^A 与 μ_u^T 直接作差, 再经过单层全连接网络来提取偏置信息, 即

$$f(\mu_u^A, \mu_u^T) = g((\mu_u^A - \mu_u^T) w_{f1} + b_{f1})$$

其中, $g(\cdot)$ 指的是一个具体的激活函数, $w_{f1} \in \mathbb{R}^{k \times k}$ 和 $b_{f1} \in \mathbb{R}^{1 \times k}$ 分别指的是权重矩阵和偏置向量。

本文将使用上述公式的模型记为 Tri-VAE-sub, 并使用了 tanh 函数和 sigmoid 函数作为激活函数进行实验, 最后的实验结果如表 5 所示。结合表 3、表 4 和表 5 的实验结果可以看出, Tri-VAE-sub 的推荐性能比 Tri-VAE 稍差一些, 但比所有基线模型都要好。这说明通过利用 μ_u^A 和 μ_u^T 之间的差异能够纠正用户的潜在特征, 进而有效缓解偏置带来的影响。另外,

表 5 在数据集 Yahoo! R3 和 Coat Shopping 上, Tri-VAE 和它的变体 Tri-VAE-sub 的推荐性能比较, 其中最佳结果用粗体标记

Table 5 Recommendation performance between Tri-VAE and its variant Tri-VAE-sub on Yahoo! R3 and Coat Shopping, where the best results are marked in bold

数据集	方法	激活函数	AUC
Yahoo! R3	Tri-VAE	tanh	0.7833
		sigmoid	0.7763
	Tri-VAE-sub	tanh	0.7801
		sigmoid	0.7751
	Tri-VAE	tanh	0.6791
		sigmoid	0.7432
Coat Shopping	Tri-VAE-sub	tanh	0.6835
		sigmoid	0.7427

对于衡量 μ_u^A 和 μ_u^T 差异的方式, 使用拼接的方法比作差的方法要好一点, 这是因为使用拼接能够最大程度地保留两者差异的信息, 更有利于神经网络对偏置信息的提取。

此外, 在 Tri-VAE 中, 特征校正模块使用了带激活函数的单层全连接网络提取偏置特征。因此, 接下来本文从网络层数的角度来进一步探讨特征校正模块, 并先后使用了双层全连接网络和三层全连接网络。本文将使用双层全连接网络的模型记为 Tri-VAE(双层), 将使用三层全连接网络的模型记为 Tri-VAE(三层)。同样地, 使用了 tanh 函数和 sigmoid 函数作为激活函数进行实验。另外, 对于同一个模型, 多层全连接网络的设置相同, 比如在实验过程中, 模型选择 tanh 函数, 则双层全连接网络都使用 tanh 函数。Tri-VAE(双层)和 Tri-VAE(三层)的推荐性能结果如表 6 所示。从实验结果可以看出, 对于数据集 Yahoo! R3 而言, 增加网络层数并不能提高模型的推荐性能, 而对于数据集 Coat Shopping 而言, 增加网络层数能够进一步提高模型的推荐性能。这可能与数据集本身包含的偏置信息有关, 不同的数据集受偏置的影响程度不同。另外还有可能与无偏数据规模的大小有关, 对于 Yahoo! R3 而言, 无偏数据的规模相对于有偏数据比 Coat Shopping 更小, 在同样的数据规模下, 增加网络层数相当于增加了模型参数, 这使得模型的训练更加困难。

表 6 在数据集 Yahoo! R3 和 Coat Shopping 上, 特征校正模块网络层数对推荐性能的影响, 其中最佳结果用粗体标记

Table 6 Recommendation performance of our proposed model with different numbers of network layers on Yahoo! R3 and Coat Shopping, where the best results are marked in bold

数据集	方法	激活函数	AUC
Yahoo! R3	Tri-VAE(单层)	tanh	0.7833
		sigmoid	0.7763
	Tri-VAE(双层)	tanh	0.7807
		sigmoid	0.7784
	Tri-VAE(三层)	tanh	0.7781
		sigmoid	0.7741
Coat Shopping	Tri-VAE(单层)	tanh	0.6791
		sigmoid	0.7432
	Tri-VAE(双层)	tanh	0.6796
		sigmoid	0.7407
	Tri-VAE(三层)	tanh	0.6763
		sigmoid	0.7511

6 总结

本文首先提出了推荐系统中一个重要且前沿的问题,即一元反馈的偏置问题,并对其进行了严格的定义和形式化。针对这一问题,从一个新颖的角度提出了一种基于多任务学习和变分自编码器的模型,即三任务变分自编码器。该模型将有偏数据、无偏数据和联合数据当作三个不同的学习任务,同时任务之间通过特征校正模块和标签增强模块进行关联。其中,特征校正模块用于获取无偏的用户潜在特征,而标签增强模块则是为了进一步利用无偏数据的信息。本文在两个真实的数据集上进行实验,实验结果表明该模型能够有效缓解偏置的影响,比现有的推荐模型有更好的推荐性能。此外,为了进一步研究所提出的模型,本文对其进行了消融实验、超参数敏感性分析和收敛性分析,同时还进一步对其中的特征校正模块进行了探讨。

在推荐系统中,保护用户的隐私越来越受学术界和工业界的关注。在未来的工作中,我们将进一步研究面向用户反馈等隐私信息的保护技术^[39],以适应用户对隐私和数据安全的需求。

参考文献

- [1] Zeng C, Xing C X, Zhou L Z. A Survey of Personalization Technology[J]. *Journal of Software*, 2002, 13(10): 1952-1961.
(曾春, 邢春晓, 周立柱. 个性化服务技术综述[J]. *软件学报*, 2002, 13(10): 1952-1961.)
- [2] Huang L W, Jiang B T, Lv S Y, et al. Survey on Deep Learning Based Recommender Systems[J]. *Chinese Journal of Computers*, 2018, 41(7): 1619-1647.
(黄立威, 江碧涛, 吕守业, 等. 基于深度学习的推荐系统研究综述[J]. *计算机学报*, 2018, 41(7): 1619-1647.)
- [3] Zhang Y J, Dong Z, Meng X W. Research on Personalized Advertising Recommendation Systems and Their Applications[J]. *Chinese Journal of Computers*, 2021, 44(3): 531-563.
(张玉洁, 董政, 孟祥武. 个性化广告推荐系统及其应用研究[J]. *计算机学报*, 2021, 44(3): 531-563.)
- [4] Chen C, Zhang M, Ma W Z, et al. Jointly Non-Sampling Learning for Knowledge Graph Enhanced Recommendation[C]. *The 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020: 189-198.
- [5] Chen J W, Dong H D, Wang X, et al. Bias and Debias In Recommender System: A Survey and Future Directions[EB/OL]. 2020.
- [6] Guo H F, Yu J K, Liu Q, et al. PAL: A Position-Bias Aware Learning Framework for CTR Prediction In Live Recommender Systems[C]. *The 13th ACM Conference on Recommender Systems*, 2019: 452-456.
- [7] Mansoury M, Abdollahpouri H, Pechenizkiy M, et al. Feedback Loop and Bias Amplification In Recommender Systems[C]. *The 29th ACM International Conference on Information & Knowledge Management*, 2020: 2145-2148.
- [8] Ji Y G, Shi C, Zhuang F Z, et al. Integrating Topic Model and Heterogeneous Information Network for Aspect Mining with Rating Bias[C]. *Advances in Knowledge Discovery and Data Mining*, 2019: 160-171.
- [9] Schnabel T, Swaminathan A, Singh A, et al. Recommendations As Treatments: Debiasing Learning and Evaluation[EB/OL]. 2016: ArXiv preprint arXiv:1602.05352.
- [10] Bonner S, Vasile F. Causal Embeddings for Recommendation[C]. *The 12th ACM Conference on Recommender Systems*, 2018: 104-112.
- [11] Liu D G, Cheng P X, Dong Z H, et al. A General Knowledge Distillation Framework for Counterfactual Recommendation via Uniform Data[C]. *The 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020: 831-840.
- [12] Yuan B W, Hsia J Y, Yang M Y, et al. Improving Ad Click Prediction by Considering Non-Displayed Events[C]. *The 28th ACM International Conference on Information and Knowledge Management*, 2019: 329-338.
- [13] Saito Y. Asymmetric Tri-Training for Debiasing Missing-not-At-Random Explicit Feedback[C]. *The 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020: 309-318.
- [14] Islam R, Keya K N, Zeng Z Q, et al. Debiasing Career Recommendations with Neural Fair Collaborative Filtering[C]. *The Web Conference 2021*, 2021.
- [15] Sun X Y, Liu H F, Jing L P, et al. Deep Generative Recommendation Based on List-Wise Ranking[J]. *Journal of Computer Research and Development*, 2020, 57(8): 1697-1706.
(孙肖依, 刘华锋, 景丽萍, 等. 基于列表级排序的深度生成推荐方法[J]. *计算机研究与发展*, 2020, 57(8): 1697-1706.)
- [16] Liang D W, Krishnan R G, Hoffman M D, et al. Variational Autoencoders for Collaborative Filtering[C]. *The 2018 World Wide Web Conference*, 2018: 689-698.
- [17] Ricci F, Rokach L, Shapira B. Recommender Systems Handbook[M]. Boston, MA: Springer US, 2015.
- [18] Luo X, Ouyang Y X, Xiong Z, et al. The Effect of Similarity Support In K-Nearest-Neighborhood Based Collaborative Filtering[J]. *Chinese Journal of Computers*, 2010, 33(8): 1437-1445.
(罗辛, 欧阳元新, 熊璋, 等. 通过相似度支持度优化基于 K 近邻的协同过滤算法[J]. *计算机学报*, 2010, 33(8): 1437-1445.)
- [19] Sarwar B, Karypis G, Konstan J, et al. Analysis of Recommendation

- tion Algorithms for E-Commerce[C]. *The 2nd ACM conference on Electronic commerce*, 2000: 158-167.
- [20] Deshpande M, Karypis G. Item-Based Top- N Recommendation Algorithms[J]. *ACM Transactions on Information Systems*, 2004, 22(1): 143-177.
- [21] Karypis G. Evaluation of Item-Based Top-N Recommendation Algorithms[C]. *The tenth international conference on Information and knowledge management*, 2001: 247-254.
- [22] Kim C, Kim J. A Recommendation Algorithm Using Multi-Level Association Rules[C]. *The 2003 IEEE/WIC International Conference on Web Intelligence*, 2003: 524-527.
- [23] Rendle S, Freudenthaler C, Gantner Z, et al. BPR: Bayesian Personalized Ranking from Implicit Feedback [EB/OL]. 2012: ArXiv preprint arXiv:1205.2618.
- [24] Kabbur S, Ning X, Karypis G. FISM: Factored Item Similarity Models for Top-N Recommender Systems[C]. *The 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2013: 659-667.
- [25] Johnson C. Logistic Matrix Factorization for Implicit Feedback Data[J]. *Advances in Neural Information Processing Systems*, 2014, 27(78): 1-9.
- [26] He X N, Liao L Z, Zhang H W, et al. Neural Collaborative Filtering[C]. *The 26th International Conference on World Wide Web*, 2017: 173-182.
- [27] Sedhain S, Menon A K, Sanner S, et al. AutoRec: Autoencoders Meet Collaborative Filtering[C]. *The 24th International Conference on World Wide Web*, 2015: 111-112.
- [28] Saito Y, Yaginuma S, Nishino Y, et al. Unbiased Recommender Learning from Missing-not-At-Random Implicit Feedback[C]. *The 13th International Conference on Web Search and Data Mining*, 2020: 501-509.
- [29] Zhu Z W, He Y, Zhang Y, et al. Unbiased Implicit Recommendation and Propensity Estimation via Combinational Joint Learning[C]. *RecSys '20: Fourteenth ACM Conference on Recommender Systems*, 2020: 551-556.
- [30] Xiaojie Wang, Rui Zhang, Yu Sun, et al. Doubly Robust Joint Learning for Recommendation on Data Missing not at Random[C]. *The International Conference on Machine Learning*, 2019: 6638-6647.
- [31] Wei T X, Feng F L, Chen J W, et al. Model-Agnostic Counterfactual Reasoning for Eliminating Popularity Bias In Recommender System [EB/OL]. 2020: ArXiv preprint arXiv:2010.15363.
- [32] Liu D G, Lin C, Zhang Z L, et al. Spiral of Silence In Recommender Systems[C]. *The Twelfth ACM International Conference on Web Search and Data Mining*, 2019: 222-230.
- [33] Zeng S X, Bayir M A, Pfeiffer J J III, et al. Causal Transfer Random Forest: Combining Logged Data and Randomized Experiments for Robust Prediction [EB/OL]. 2020: ArXiv preprint arXiv:2010.08710.
- [34] Wang X J, Zhang R, Sun Y, et al. Combating Selection Biases In Recommender Systems with a few Unbiased Ratings[C]. *The 14th ACM International Conference on Web Search and Data Mining*, 2021: 427-435.
- [35] Schnabel T, Bennett P N. Debiasing Item-to-Item Recommendations with Small Annotated Datasets[C]. *RecSys '20: Fourteenth ACM Conference on Recommender Systems*, 2020: 73-81.
- [36] Kingma D P, Welling M. Auto-Encoding Variational Bayes [EB/OL]. 2013: ArXiv preprint arXiv:1312.6114.
- [37] Danilo Jimenez Rezende, Shakir Mohamed, Daan Wierstra. Stochastic Backpropagation and Approximate Inference in Deep Generative Models[EB/OL]. 2014: ArXiv preprint arXiv:1401.4082.
- [38] Marlin B M, Zemel R S. Collaborative Prediction and Ranking with Non-Random Missing Data[C]. *The third ACM conference on Recommender systems*, 2009: 5-12.
- [39] Li X G, Li H, Li F H, et al. A Survey on Differential Privacy[J]. *Journal of Cyber Security*, 2018, 3(5): 92-104.
- (李效光, 李晖, 李风华, 等. 差分隐私综述[J]. *信息安全学报*, 2018, 3(5): 92-104.)



林子楠 于 2020 年在深圳大学自动化专业获得学士学位。现在深圳大学计算机科学与技术专业攻读硕士学位。研究领域为推荐系统和迁移学习。Email: lzn87591@gmail.com



刘杜钢 于 2019 年在厦门大学计算机专业获得硕士学位。现在深圳大学计算机科学与技术专业攻读博士学位。研究领域为推荐系统、反事实机器学习和因果推理。Email: dugang.ldg@gmail.com



潘微科 于 2012 年在香港科技大学计算机科学及工程专业获得博士学位。现任深圳大学计算机与软件学院副教授。研究领域为数据挖掘和人工智能。研究兴趣包括: 迁移学习、联邦学习、推荐系统和机器学习。
Email: panweike@szu.edu.cn



明仲 于 2003 年在中山大学计算机科学与理论专业获得博士学位。现任深圳大学计算机与软件学院教授。研究领域为软件工程和人工智能。研究兴趣包括: 软件中间件、形式化方法、推荐系统和机器学习。
Email: mingz@szu.edu.cn