

文章编号:2095-7386(2020)01-0062-06

DOI:10.3969/j.issn.2095-7386.2020.01.012

基于 Adam 的推荐系统模型研究

樊翔宇,张 聪

(武汉轻工大学 数学与计算机学院 湖北 武汉 430023)

摘 要:传统技术实现的推荐系统探索用户兴趣时十分依赖用户的历史行为等额外数据,探索过程复杂且面临严重的冷启动问题。为降低对用户信息的依赖,同时减轻冷启动问题,采用当前人工智能中先进的 Adam 优化方法提出了一种推荐系统模型。使用 Tensor flow 平台在一系列数据集上的仿真结果表明,与其他模型相比,提出的模型显著降低了对用户信息的需求,具有更精准的推荐结果,在应对冷启动问题时也有较好的性能表现。

关键词:推荐系统;冷启动;人工智能;模型优化

中图分类号:TP 39

文献标识码:A

Research on recommendation system model based on adam

FAN Xiang-yu, ZHANG Cong

(School of Mathematics and Computer Science, Wuhan Polytechnic University, Wuhan 430023, China)

Abstract: Recommendation system implemented by traditional technology relies heavily on user's historical behavior equivalent data when exploring user interest, meanwhile the user interests exploration process is complex and faces serious cold start problems. In order to reduce the dependence on user information and optimizing the cold start problem, a new recommendation system model was proposed by using the advanced Adam optimization method in the current artificial intelligence. The simulation results on a series of datasets using Tensor flow platform show that the proposed model significantly reduces the dependency for user information, and has more accurate recommendation results and better performance in dealing with cold start problems compared with other models.

Key words: recommendation system; cold-start; artificial intelligence; model optimization

1 引言

在现代社会,人们每天需要接受和处理的各类信息越来越多,而对于特定人类个体来说,很多接触到的信息其实都是该个体不感兴趣的或者说是无用的信息,而目前很多商业模式往往又十分依赖向用户推荐信息的方式运作,推荐系统推荐效果的好坏直接影

响着这些商业模式的经营效果。这就引申出一个怎样向特定个体精准推送其感兴趣的信息的问题。

David Goldberg 等人首先提出了协同过滤(Collaborative Filtering)的方法解决推荐系统的相关问题^[1]。随后 Andrew I. Schein 等人对冷启动问题展开了研究,他们提出了在单个概率下应对冷启动问题的方法^[2],这在一定程度上减轻了单个概率条

收稿日期:2019-10-26.

作者简介:樊翔宇(1996-),男,硕士研究生,E-mail:493366900@qq.com.

通信作者:张聪(1968-),男,教授,博士.

基金项目:国家自然科学基金面上项目(61272278);湖北省重大科技专项(2018ABA099).

件下推荐系统面临的冷启动问题。伴随移动互联网的发展,社交网络开始在人类社会中大范围应用,Sedhain 等人适时地尝试了将用户在社交网络中的行为数据引入到推荐系统中^[3-5],借助这些数据在冷启动场景下取得了较好的推荐效果。但在实际应用中,一方面获取用户的社交网络信息面临着隐私保护的问题^[6],另一方面更多信息的引入也进一步加深了推荐系统对用户行为信息的依赖,进而导致系统对用户兴趣探索的复杂度也越来越高。

近年来人工智能技术不断兴起,在很多领域都带来了颠覆性的革新,取得了很好的效果。Hao Wang 等人将人工智能中的深度学习等技术引入到推荐系统领域的相关研究中^[7],初步展现出了人工智能技术应用在推荐系统领域的良好前景。

时至今日,人们通过网络进行在线娱乐活动变得越来越普遍和频繁,减少用户花费在自己不感兴趣的事情上的时间,提高用户获取感兴趣的信息的效率逐渐演变成一个亟待解决的问题。观看电影作为一种耗时相对较长的社会活动,十分适合作为进行推荐系统研究的场景,研究成果也易于向其他系统中迁移。在这一背景和前人研究的基础上,尝试提出了一种新的基于人工智能技术的推荐系统模型,并使用 Tensor flow 平台在 movie lens 和 jester 两个数据集上进行了验证实验,模型取得了较好的推荐效果,在冷启动环境下亦具备一定的可用性。此外理论分析表明在大部分推荐系统经常遇到的“用户—评分”类应用场景下,模型具有一定的通用性。

2 主流推荐算法及不足

为便于论述,首先介绍几种推荐系统领域主流的算法,指出其原理和缺陷。

2.1 主流推荐系统算法

在人工智能技术出现之前,协同过滤算法是推荐系统领域最常用的模型构建算法,而协同过滤算法又分为基于内容的协同过滤和基于用户的协同过滤两种类型。基于内容相似度时其算法核心思想是首先在整个系统中寻找该用户已经表示喜好的内容之外的其他内容,且这些内容应当按照与用户已表示喜好的内容的相似度从大到小排序,最后从内容序列中取出前 N 个结果推荐给用户;基于用户相似度时其算法核心思想与基于内容相似度时类似,不同之处在于算法会首先根据当前用户的喜好在系统内所有其他用户中寻找与当前用户最相似的若干

个用户,对找到的用户按相似度从大到小排序,随后将这些用户所喜好的内容也按照喜好程度从大到小排序,最后从得到的序列中取出前 N 个内容为用户做推荐。

Slope One 算法^[8]及其系列改进^[9]也是推荐系统领域主流的算法之一。如果需要给出某个用户对特定条目的评分,该算法会首先搜索所有对这个条目都评过分的用户,再计算这些用户对所有共同评分的差值,依据差值给出具体推荐。

2.2 主流方法的缺陷

朴素的协同过滤方法只有在系统中存在大量用户信息和内容信息且同时拥有众多用户历史行为记录的情况下才能取得较好的推荐效果。即使已经出现了许多新方法和手段^[10],这种对用户历史行为信息的依赖依然给推荐系统带来了难以解决的冷启动问题。

使用 Slope One 算法搭建推荐系统,要求用户已经对作为参考的内容给出了评分,这显然是一种理想假设,不是任何情况下都能满足的。

3 模型及相关改进

模型的主要思想是将人工智能领域中当前比较先进的 Adam 迭代优化方法引入到推荐系统中,以“用户—评分”矩阵为参照优化一个损失函数,进而根据优化结果为用户完成推荐。

3.1 Adam 优化器

Adam 优化器实际上是对 adaptive moment estimation 优化算法的封装。在 Adam 算法出现之前,已经有两种与之有渊源的基于梯度下降思想的优化算法在生产和研究中得到了广泛的应用,分别是 RMS prop (root mean square propagation) 算法和 Ada-grad (adaptive gradient) 算法^[11-12]。Adam 算法结合了 RMS prop 算法和 Ada-grad 算法的优点,因此相对 RMS prop 算法和 Ada-grad 算法而言,采用 Adam 算法优化得到的模型的实际应用效果在大多数场景下都要更好。

3.2 多维数据归约

为了减少训练过程中的时间开销和空间开销,提升计算速度,使建立的模型具有在生产环境中使用的可行性,模型在定义损失函数时包含了一部分对数据进行归约处理的操作。所谓数据归约,指的是在尽可能保持数据原貌的情况下,最大限度地精简数据量^[9],而多维数据归约是在多个维度上对数

据进行归约处理,一般包括特征归约、样本归约和特征值归约。具体来说,特征归约对数据中不重要或者是相关度不高的特征做删除处理,从而减少特征向量的维度;样本归约从数据集中选出一部分具有代表性的样本组成数据集的一个子集,从而达到减少计算成本的目的;特征值归约所做的工作是将连续的特征值离散化,并将离散之后的结果映射到多个小区间中,采用离散的符号来表示每一个小区间,最终简化数据的描述。

3.3 模型概述

模型首先对原始数据进行预处理,清洗其中的异常数据,接着使用处理后的数据构建“用户—评分”矩阵,随后对矩阵中每个物品的所有评分分别求均值,再对得到的结果按公式(1)进行归一化处理,式中 m 表征用户打分的个数, t_i 指的是用户打分的分值。

$$N_{(i)} = t_i - (\sum_{i=1}^m t_i)/m. \quad (1)$$

得到归一化的结果之后,模型首先根据数据集的特征选定维度,并生成两个分别与数据集中用户数量和数量相等的符合正态分布的数值集合以拟合用户和内容,在得到的两个数值集合的辅助下定义推荐系统模型损失函数,函数由公式(2)给出,其中 i 是用户索引, j 是内容索引, u 表示所有用户总和, x^j 是特征, n 为待提取的维度, λ 为由 Adam 优化决定的参数。

$$j(\theta) = \frac{1}{2} \sum_{j=1}^u \sum_{i,r(i,j)=1} ((\theta^{(j)})^T x^i - y^{(i,j)})^2 + \frac{\lambda}{2} \sum_{j=1}^u \sum_{k=1}^n (\theta_k^{(j)})^2. \quad (2)$$

以上步骤完成之后,使用 Adam 优化器优化建立的模型,根据得到的两个矩阵预测每个用户对所有物品的评分的估计值,预测方法见公式(3),其中 C 、 U 是两个矩阵,分别表示内容和用户, T 表示转置操作, $*$ 表示点乘, $\bar{c}_i$ 是所有用户对当前待处理项目的打分均值,将预测值组合成评分预测矩阵。至此只要向模型中输入指定的用户标识,模型即可根据评分预测矩阵为用户做出对应的推荐。总的来说模型需要的输入数据仅仅是一个“用户—评分”矩阵,对其它数据并没有迫切的需求,由此可以看出模型减少了对用户数据的依赖;此外由于辅助数值集合的引入,在应对高度稀疏矩阵时,模型的性能仍具有较好的可用性。

$$P_i = C_i * U_i^T + \bar{c}_i. \quad (3)$$

根据以上叙述,现给出如图 1 所示的模型结构图。

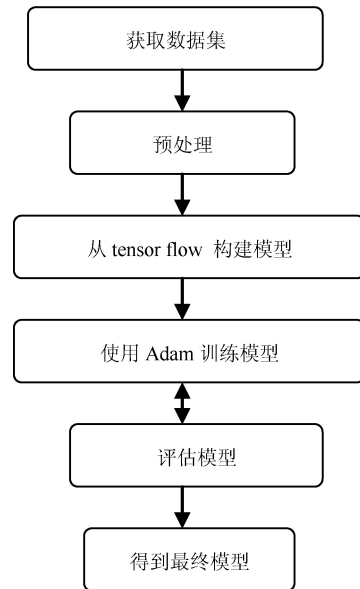


图 1 模型流程图

值得特别注意的是,在将模型应用到一个新的数据集或推荐系统上时,训练模型和评估这两个阶段一定不能割裂,在每一轮模型训练结束后,应当根据评估结果动态调整生成辅助数值集合时选择的相关参数,直至模型在即将应用的数据集上取得较好的评估效果为止。向一个已经搭建好的推荐系统中加入新的用户或新的内容,这种行为称为推荐系统的冷启动,由于新用户或新内容缺少用户行为支撑,常规推荐系统中因此带来的对新用户或新内容相似度等属性计算的难题即为推荐系统面临的冷启动困境,而上述提出的推荐系统面临冷启动场景时,只要根据新加入的数据重新训练和评估模型,模型就可以继续投入使用,因此模型亦可应用于冷启动环境。

4 实验及结果分析

为验证所提出模型的可行性和效果,使用 movie lens 数据集和 jester 数据集进行实验,同时为详细论述实验部分,先对实验采用的 Tensor flow 框架和相关数据集进行一个简介。此外选取平均绝对误差 (MAE) 为实验的评估标准,计算方法如下:

$$MAE = \frac{1}{m} \sum_{i=1}^m |x^{(i)} - \bar{x}^{(i)}|. \quad (4)$$

式中 m 为所有对象总数, $x^{(i)}$ 是真实结果, $\bar{x}^{(i)}$ 是预测结果。

4.1 Tensor flow 框架

Tensor flow 是一种机器学习系统,可在大规模和异构环境中运行^[13]。Tensor flow 框架内部使用张量 (Tensor) 作为基本的运算单位,张量可以看作

一种 N 维数组,也可以理解为一种 N 维向量,图 2 自左至右分别给出了单维、二维和三维的张量示例。张量中包含的维数理论上是可以无限扩展的,张量的这一特性有效降低了在 Tensor flow 框架中采用人工智能相关的算法对海量数据进行处理难度,提供了在 Tensor flow 框架上搭建推荐系统的环境基础。

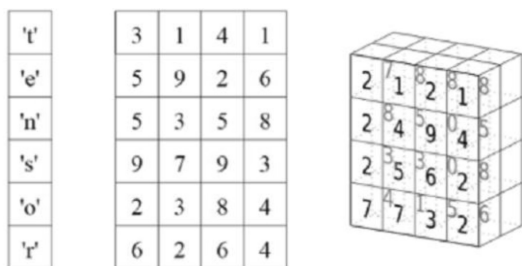


图 2 张量(tensor)

很多人工智能算法需要不断迭代优化和海量数据支撑才能展现良好的效果,因此对机器计算能力有很高要求。流(flow)作为 Tensor flow 最重要的特性之一,为 Tensor flow 带来了分布式和并行计算。流实际上指数据流图,类似传统数据结构中的图数据结构,数据流图也由若干个节点构成,每个节点包含一部分独立的计算任务。在集群计算环境中,这些节点会被分别映射到集群中不同的处理器内核上,即使在单机多核环境下 Tensor flow 也可以高效地进行映射,如果系统中配置有一个或多个 GPU,数据流图中的节点也可以被映射到 GPU 系统中的不同内核上。图 3 上边是单机环境下的映射,下边是机器集群环境下的映射。

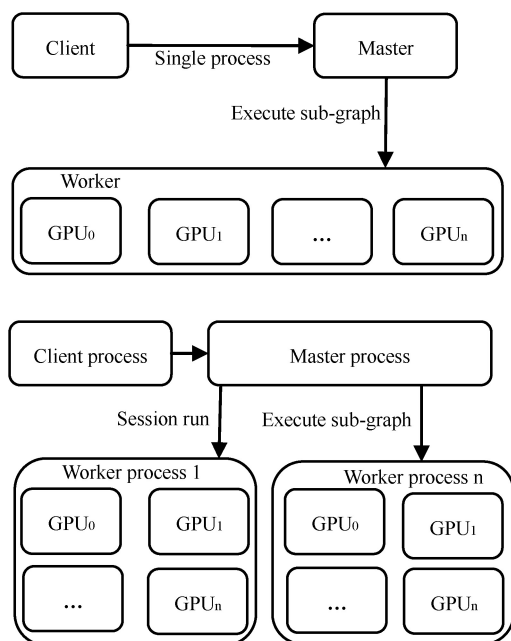


图 3 流和集群映射

Tensor flow 中多样的 API 也是不得不提的一个重要特性。为简化人工智能相关领域的数据处理、模型训练和结果导出之类的工作,使研究者能够专注对模型和算法本身的研究及优化,Tensor flow 提供了适用于不同级别的 API 接口。张量和数据流图就属于低级别的 API,此外 Tensor flow 中启动会话的 Session 也属于这一级别。高级别 API 包括估算器(Estimator)、实验类(Experiment)和数据集(Dataset)等。估算器封装了模型参数配置和模型评估等常用的机器学习方法;实验类提供了常规的训练模型的方法,与估算器互为补充;数据集 API 封装了对数据进行操作的基本工具。可以说正是因为提供了这些高效简洁的 API,使 Tensor flow 变得易用和流行。

4.2 movie lens 和 jester 数据集

movie lens 电影数据集共包含三个数据文件,分别是用户信息集、用户评分集和电影信息集,模型只使用其中的用户评分集和电影信息集。在用户评分集中包含有用用户 id、电影 id、用户对电影的评分,以及用户评分时的时间戳,用户的打分区间为 $[0,5]$;电影信息集中包含电影 id、电影名称、电影分类,电影名称中还嵌入了电影上映的年代等信息。

选取的 jester 数据集是一个笑话评分数据集,选择的是 Ken Goldberg 团队在其官网上最新开源的版本,这一版本保证每个笑话都存在评分记录。

4.3 验证实验

对选取的数据集依照 90% 训练数据和 10% 测试数据做分割;接下来建立全 0 的“用户—评分”矩阵,根据载入的评分数据初始化矩阵内容;按给出的归一化公式(1)对矩阵中的原始评分进行归一化处理;引入 Tensor flow 框架,使用框架提供的 random_normal() 函数生成模型需要的服从指定正态分布的数值集合。预研实验的结果表明,对于 movie lens 数据集和 jester 数据集,设定模型的 stddev 值为 1, features 值为 12 是较为合适的选择,其中 stddev 指的是所用正态分布的标准差,features 指的是选取的数据特征的维数;根据给出的模型损失计算公式(2)定义模型的损失计算函数,并在函数中调用 Tensor flow 提供的 reduce_sum() 函数对数据进行归约处理。至此,模型的构建工作基本完成。

首先进行验证实验以证实模型在现实世界的可行性。使用 Adam 优化器对文中建立的模型进行 8 000 步迭代优化,设定初始学习率为 0.001,算法需要的其他参数由优化器自行选择,两个数据集的迭

代优化曲线如图 4 所示,其中左边为在 movie lens 数据集上的结果,右边为在 jester 数据集上的结果,横轴为训练步数,纵轴为模型损失。

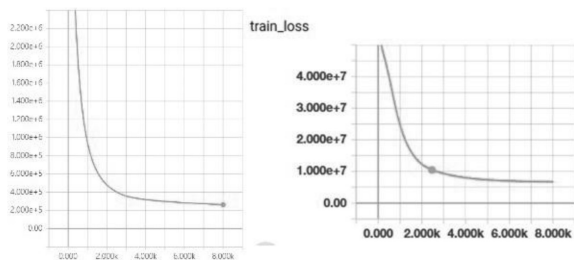


图 4 步数—损失曲线图

从图 4 可以看出经 8 000 次迭代时模型对于两个数据集的损失均已经趋于收敛,继续加大迭代步数会显著加大计算开销,却无法获得与之匹配的迭代收益,甚至有可能带来过拟合问题,实验证明迭代步数设为 8 000 是较为合理的选择。

使用平均绝对误差在测试数据集上对优化之后的模型进行评估,结果如表 1 所示。

表 1 实验结果汇总(越小越好)

数据集/算法	Item CF	User CF	新模型
movie lens	0.86	1.04	0.70
jester	4.57	9.12	4.22

向 movie lens 数据集中加入 10 名模拟生成的新用户拟合冷启动场景,重新训练模型并在测试数据集上随机加入 200 条符合约束的新用户评分记录,再次使用平均绝对误差在测试数据集上评估后得到的 MAE 值增加至 0.701,增加幅度约为 0.3%,实验结果表明虽然在冷启动场景下模型的性能有所降低,但依旧处于可接受范围内。

随机选取 1 名用户,使用模型在 movie lens 数据集上对这两名用户进行 Top 10 推荐,推荐系统的输出结果如图 5 所示。

综合验证实验的结果,提出的模型可以较好地工程上实现,能够较为有效地给出推荐结果,在现实世界中具有一定的可行性。

为用户 3929 推荐的 10 部电影是:

预测评分: 4.46 Reds (1981)
 预测评分: 4.35 Beetlejuice (1988)
 预测评分: 4.23 Kiss of Death (1995)
 预测评分: 4.17 Aladdin (1992)
 预测评分: 4.15 Princess Bride, The (1987)
 预测评分: 4.12 Brazil (1985)
 预测评分: 4.12 Sound of Music, The (1965)
 预测评分: 4.07 Celestial Clockwork (1994)
 预测评分: 4.04 Goodbye Girl, The (1977)
 预测评分: 4.04 Walk on the Moon, A (1999)

图 5 推荐结果

4.4 对比实验

为进一步验证模型的性能,另行选择了一些经典的推荐系统算法与模型进行对比实验。进行对比实验时,为最大限度地保持公平性,所选择的其他算法在判断相似度时统一采用皮尔逊相关系数作为判定标准;在划分数据集时,所有对比实验使用的测试数据集索引和验证数据集索引均与模型验证实验保持一致,即训练数据集所占比例为 90%,测试数据集为 10%,且划分数据集时的随机数种子也与模型验证实验保持一致。皮尔逊相关系数计算方法见公式(5), x, y 分别表示待计算对象, E 指期望。

$$\rho_{x,y} = \frac{E(x,y) - E(x)E(y)}{\sqrt{E(x^2) - E^2(x)} \sqrt{E(y^2) - E^2(y)}} \quad (5)$$

此外,由于现有可用于对比的算法在冷启动状态下并不能很好的适应,因此在本文中笔者不再进行冷启动状态下的对比实验,但会从理论上给出预测和分析。

4.4.1 实验一

第一个对比实验采用基于内容相似度的协同过滤算法(Item CF),两种算法推荐准确度的对比如表 1 所示。

从理论上来说,基于内容的协同过滤算法十分需要用户行为记录支撑算法性能,在面对高度稀疏的评分矩阵时推荐效果将严重下降直至算法无法运行,这也是基于内容的协同过滤比起所提模型另一个主要的不足之处,因此在冷启动状态下,失去行为信息的算法势必不会有太好的预期推荐精度。

4.4.2 实验二

第二个对比实验选取基于用户相似度的协同过滤算法(User CF),与所提模型推荐准确度的对比结果如表 1 所示。

从理论上分析实验结果,基于用户的协同过滤算法需要额外的用户信息计算用户之间的相似度,比所提模型对用户信息的依赖程度更高,因此基于用户的协同过滤算法推荐准确度相比所提模型在这一方面存在劣势,其性能表现甚至不如基于内容的协同过滤算法。

4.5 结论

验证实验证明了模型在现实世界中的可行性,对比实验证明了相比当前常见的其他推荐系统算法,新模型的推荐准确度在两个数据集上都有优势;同时实验还模拟了冷启动环境,并从理论上分析了新模型在冷启动场景下相比其他主流算法更广泛的

适用性,对比实验的结果和理论分析表明相对其他主流算法,模型在对用户信息的依赖程度较低的同时,还拥有更好的推荐准确性和冷启动场景适用性。

5 结束语

笔者结合人工智能领域的相关优化技术提出了一种用于推荐系统的模型,并将提出的模型在 Tensor flow 框架上做了工程实现。相比其他主流推荐算法,提出的模型在推荐准确度上具有一定的优势,拥有更佳的推荐准确度;此外模型减少了主流推荐算法对用户额外信息的依赖,在“用户—评分”类应用环境下稍加更改即可直接应用;实验和理论分析证明面对冷启动环境时模型的性能未见显著下降,仍然处于可用状态。更为重要的是模型可较为简单地使用 Tensor flow 框架实现,由于有了优化器的帮助,对模型进行优化也不需要过多的人工干预和调整,模型在工程和生产环境中应用和更替时的开销较小,因此模型尤其适合应用在生产环境中。

在今后的工作中,将继续着重解决模型在面对高维极度稀疏矩阵时性能下降的问题,同时将尝试在更多的数据集上应用模型及考虑根据实际情况为模型添加预测条件约束以进一步改进模型的性能,避免模型中可能存在的溢出问题,继续提高模型的通用性,将模型拓展到更多的应用场景。

参考文献:

- [1] David Goldberg, David Nichols, Brian M Oki, et al. Using collaborative filtering to weave an information tapestry [J]. Communications of the ACM, 1992, 35(12): 61-70.
- [2] Schein A I, Popescul A, Ungar L H, et al. Methods and Metrics for Cold-Start Recommendations [C]. Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2002: 253-260.
- [3] Sedhain S, Sanner S, Braziunas D, et al. Social collaborative filtering for cold-start recommendations [C]. Proceedings of the 8th ACM Conference on Recommender systems, 2014: 345-348.
- [4] 陈克寒, 韩盼盼, 吴健. 基于用户聚类的异构社交网络推荐算法 [J]. 计算机学报, 2013, 36(02): 349-359.
- [5] 宾晟, 孙更新. 基于多关系社交网络的协同过滤推荐算法 [J]. 计算机科学, 2019, 46(12): 56-62.
- [6] 王海艳, 陆金祥. 面向群组推荐的个性化隐私保护方法 [J]. 通信学报, 2019, 40(09): 106-115.
- [7] Hao Wang, Naiyan Wang, Dit-yan Yeung. Collaborative Deep Learning for Recommender Systems [C]. KDD '15 Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015: 1235-1244.
- [8] Lemire D, Maclachlan A. Slope One Predictors for Online Rating-Based Collaborative Filtering [C]. Proceedings of the 2005 SIAM International Conference on Data Mining, SDM 2005.
- [9] 沈学利, 赫辰皓, 孟祥福. 受限玻尔兹曼机与加权 Slope One 的混合推荐算法研究 [J]. 计算机应用研究, 2020, 37(03): 684-687.
- [10] Xiao-Yu Huang, Bing Liang, Wubin Li. Online collaborative filtering with local and global consistency [J]. Information Sciences, 2020, 506.
- [11] Ruder S. An overview of gradient descent optimization algorithms [EB/OL]. (2016-9-15) [2017-6-15] <https://arxiv.org/abs/1609.04747>, 2016.
- [12] 张慧. 深度学习中优化算法的研究与改进 [D]. 北京: 北京邮电大学, 2018.
- [13] Abadi, Martín, Barham P, Chen J, et al. TensorFlow: A system for large-scale machine learning [C]. OSDI '16 Proceedings of the 12th USENIX conference on Operating Systems Design and Implementation, 2016: 265-283.