

文章编号:1007-5321(2019)06-0035-08

DOI:10.13190/j.jbupt.2019-164

基于深度学习的融合多源异构数据的推荐模型

冀振燕, 宋晓军, 皮怀雨, 杨春

(北京交通大学 软件学院, 北京 100044)

摘要: 为了充分利用多源异构数据所提供的信息提高推荐准确度, 提出一个基于深度学习的混合推荐模型. 该模型融合评分、评论和社交网络数据进行推荐, 采用深度学习对文本和评分进行特征学习, 然后使用社交网络对采样进行约束, 从而得到更准确的用户和物品的特征表示. 实验结果表明, 该方法具有较高的准确度.

关键词: 多源异构数据; 深度学习; 推荐模型; 社交网络

中图分类号: TP311 文献标志码: A

Recommended Model for Fusing Multi-Source Heterogeneous Data Based on Deep Learning

Ji Zhen-yan, Song Xiao-jun, Pi Huai-yu, Yang Chun

(School of Software Engineering, Beijing Jiaotong University, Beijing 100044, China)

Abstract: Considering that Internet information today is diverse and inconsistent in structure, in order to fully utilize the information provided by multi-source heterogeneous data to improve the recommendation accuracy, a hybrid recommendation model based on deep learning was proposed. The model makes a recommendation based on combining ratings, review texts and social network data. The model also adopts deep learning to learn features of reviews and ratings, and then uses social network to constraint sampling. Experiments show that the model is of higher accurate feature representations of users and items.

Key words: multi-source heterogeneous data; deep learning; recommendation model; social network

为解决互联网信息过载问题, 推荐系统广泛应用于电子商务、新闻资讯、电影和音乐等网站, 其已成为互联网应用的重要组成部分. 个性化推荐系统通过对所收集的用户浏览、点击和评价等信息进行分析, 可以获取用户偏好, 从而为不同用户推荐不同物品. 随着深度学习在图像和音频识别、文本分类、表示学习等领域的成功应用, 基于深度学习的推荐系统也成为当前的研究热点.

融合多源异构数据的模型可分为推荐算法融合和数据特征融合. 推荐算法融合根据数据类型选择合适的推荐算法, 如基于内容的推荐算法和协同过

滤推荐算法融合, 协同矩阵分解算法和基于信任传播的算法融合等. 不同推荐算法之间可以采用加权、级联、混合等融合方式. 这种模型可扩展性差, 添加新的异构数据时, 需要重新设计模型.

目前, 深度学习模型在图像、文本等特定数据的表示学习中已取得理想效果, 可避免复杂的特征工程, 得到异构数据的非线性多层次的抽象特征表示, 克服多种数据的异质性. 得益于深度学习在表示学习方面的卓越效果, 基于特征融合的研究受到广泛的关注, 采用深度学习得到的不同异构数据表示可通过平均和串联等方式融合成为混合特征.

收稿日期: 2019-07-30

基金项目: 国家自然科学基金重点项目(S19A200010); 国家重点研发计划项目(R19B5200010)

作者简介: 冀振燕(1970—), 女, 副教授, 硕士生导师, E-mail: jzhenyan@hotmail.com.

提出了一个基于深度学习的推荐模型,可融合评分、评论和社交网络数据进行更准确地推荐. 模型采用基于深度学习的文本段落表示学习方法,设计了基于评分学习用户和物品特征的神经网络,并通过社交网络约束 pairwise 的学习,提高了推荐的准确度,且模型具有弹性,可扩展,可方便融合新的异构数据.

1 相关工作

近年来,深度学习在文本、图像的代表学习以及分类等领域取得突破性进展^[1],其准确度优于传统机器学习算法.

为提高基于评分数据的推荐准确度,近年来有学者将神经网络和矩阵分解相结合,提出神经网络矩阵分解(NNMF, neural network matrix factorization)^[2]. 还有学者将神经网络与协同过滤相结合提出神经协同过滤(NCF, neural collaborative filtering)^[3]. Li 等^[4]采用去噪自动编码器(denoising auto-encoders)来进行推荐. Zheng 等^[5]提出基于协同过滤的神经自回归方法. 在基于内容的推荐算法中,Elkahky 等^[6]基于深度结构化语义模型,分别学习用户和物品特征,并计算用户和物品的相似度,基于相似度确定推荐的物品. Zheng 等^[7]基于评论和评分数据,采用2个平行的神经网络分别学习用户发出的评论和物品收到的评论,然后使用一个共享层结合这2个神经网络,预测用户对物品的评分. 深度学习获取的非线性特征表示可更好地描述用户和物品的特征^[8]. 基于评论数据的推荐算法会遇到数据稀疏问题,因为每种方案都需要大量的用户评论对模型进行训练. 除了基于内容的推荐算法,深度学习还应用于协同过滤推荐算法中,包括基于受限玻尔兹曼机、自编码器、分布式表示技术、循环神经网络、生成对抗网络和其他深度学习方法的协同过滤模型. 这些算法采用深度学习技术,学习用户和物品的特征向量表示. 一般采用 pairwise 损失函数或 pointwise 损失函数,对网络参数进行训练优化,利用得到的用户和物品的向量表示进行推荐. 此外,深度学习还应用于混合推荐系统、基于社交网络的推荐系统和基于情景感知的推荐系统等^[2].

研究发现,基于深度学习的单源或者双源数据推荐已经取得了较好的结果,但是依然不能满足对推荐结果准确度的要求,那么如何使用深度学习融合多源异构数据,是值得研究的问题.

2 融合多源异构数据的推荐模型

评分数据明确但简单,评论数据语义丰富,社交网络数据反映用户彼此之间的影响,3种数据各具优势. 虽然多源异构数据可提供更丰富的信息,但不同种类的数据需要采用不同算法,算法融合较困难^[9].

随着深度学习的发展,文本等多源异构数据通过特定的深度学习神经网络均可得到较为精确的表示,并且通过神经网络自动选择有效的特征,使得异构数据特征的直接融合成为可能. 为充分利用3种数据的优点,提高推荐准确度,融合评分、评论特征,并将社交信息加入深度模型的训练过程中,提出了基于深度学习的多源异构数据推荐模型.

模型采用基于深度学习的文本段落表示学习方法——句向量的分布词袋模型(PV-DBOW, distributed bag of words version of paragraph vector)来学习评论文本的特征表示,PV-DBOW 假定文本中的单词相互独立且无序,并使用文本向量表示来预测文本中采样得到的单词. 对于评分数据,传统的矩阵分解方法面临数据稀疏和准确率较低的困难,使用神经网络提取评分特征,可更好体现用户和物品的特征. 对于社交网络数据,在 pairwise 学习方法上增加了用户的社交关系信息,使得该方法的采样更加合理,提高了推荐结果的准确度.

推荐模型的流程如图1所示. 通过深度模型学习多种数据的向量表示,然后通过串联的方法得到用户或物品的融合特征. 评论特征体现了用户对物品的态度,也可用来表示物品的特点. 模型通过 PV-DBOW 算法学习评论段落的特征表示,平均段落向量得到用户或物品的向量表示. 评分特征是对用户对物品的整体评价,体现了用户对物品的满意度. 参考 NCF 提出的网络结构,使用多层全连接的神经网络直接学习得到用户和物品的特征向量表示. 社交网络体现了用户间的好友关系,间接影响了用户和物品之间的交互,利用社交网络关系可以加强对用户潜在购买行为的预测. 使用贝叶斯个性化推荐模型(BPR, Bayesian personalized ranking)对用户和物品的非线性特征进行学习排名,从而进一步提高推荐结果准确度.

2.1 推荐模型算法

为构建一个统一的基于多源异构数据的深度学习模型,首先,构建基于用户和物品三元组的优化模

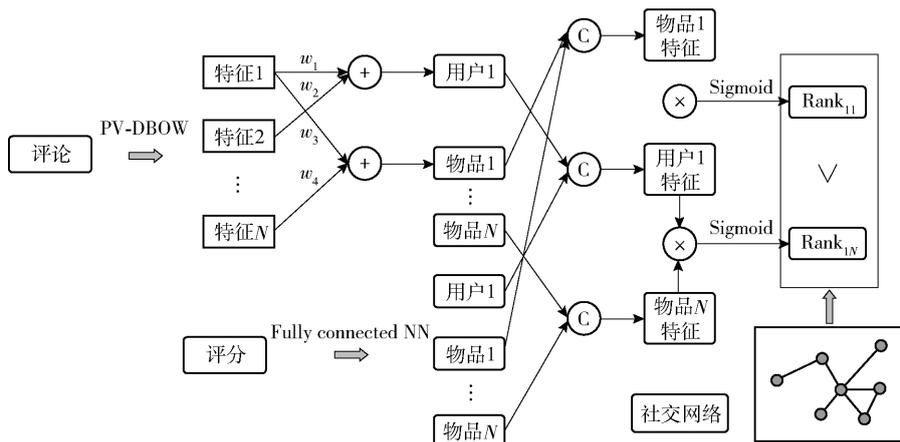


图 1 推荐模型流程

型;其次,构建用户和物品的特征,这些特征受到用户对物品的评论和评分的影响;最后,采用社交网络对用户和物品的三元组构建进一步约束,从而得到较好的推荐结果。

构造一个基于用户偏好的三元组 (u, i, j) , 该三元组代表的是用户 u 对于物品 i 的购买概率要大于物品 j 。用户三元组的构建包括 3 种情况:第 1 种是用户 u 购买了物品 i 但是没有购买物品 j , 显然用户 u 对于物品 i 的购买概率要大于物品 j ;第 2 种是用户 u 既没有购买物品 i 又没有购买物品 j , 那么无法判断用户倾向于购买哪个物品;第 3 种是用户既购买了物品 i 又购买了物品 j , 那么无法判断用户倾向于购买哪个物品。这样的分类有 2 个优点:第 1 个优点是在学习的过程中既考虑到了用户购买的物品又考虑了用户没有购买的物品, 而用户没有购买的物品就是将来需要进行排序的物品。拿出一部分未购买物品作为测试集, 保证了用户学习的数据和测试的数据没有交集。第 2 个优点是, 该方法是通过实际数据的三元组构造, 对需要排序的序列直接优化, 这样从结果的角度来说具有很强实用性。而采用评分预测的方法, 最后得到的评分对所需的排序结果之间还有一定的差距。采用基于 BPR 的方法能够直接优化得到物品的排序列表, 结果较为准确。三元组生成过程如图 2 所示。

为了求得对于用户 u 的物品推荐列表, 根据贝叶斯公式, 即为最大化以下后验概率:

$$p(\Theta|(u, i, j)) \propto p((u, i, j)|\Theta)p(\Theta) \quad (1)$$

其中: (u, i, j) 表示用户和物品的关系, Θ 表示模型的参数。通过调节模型参数, 使得实际的 (u, i, j) 用户和物品三元组出现的概率最大, 此时的参数即为

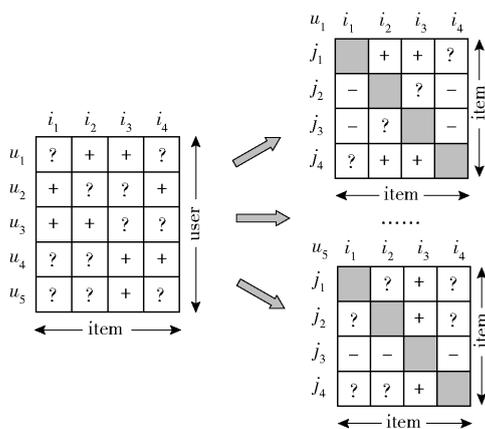


图 2 三元组生成过程

所求的模型参数。假设用户之间是相互独立的, 物品对 (i, j) 之间也是相互独立的, 那么对于所有的用户都有以下公式:

$$\prod_{u \in (u, i, j)} p((u, i, j)|\Theta) = \prod_{u \in (u, i, j)} p(i > j|\Theta) \times \prod_{u \in (u, i, j)} (1 - p(i > j|\Theta)) \quad (2)$$

其中用 (u, i, j) 来表示构造的有明确用户偏好的三元组, 对于所有属于该三元组的用户, 其购买过的物品出现的概率要大于未购买物品出现的概率。所有用户偏好三元组相乘的概率就等于属于此三元组的用户和物品的概率的阶乘再乘以不属于三元组的用户和物品概率的阶乘。

根据 pairwise 学习的整体性和反对称性, 上述公式可以被简化为

$$\prod p((u, i, j)|\Theta) = \prod_{u \in (u, i, j)} p(i > j|\Theta) \quad (3)$$

通过上面的推导, 目前还不能求得最终的排序, 需要构建一个计算模型来计算每个物品推荐的概率。使用 sigmoid 函数构建一个用户 u 购买物品 i 的

概率大于购买物品 j 的概率的模型:

$$p((u, i, j) | \Theta) := \sigma(x_{uij}(\Theta)) \quad (4)$$

其中 $x_{uij}(\Theta)$ 为一个任意的参数模型,能够描述用户和物品之间的潜在关系.也就是说,可以使用任意一个用来描述用户和物品关系的模型,结合 BPR 模型来直接优化得到一个基于物品推荐排序的推荐结果.但是基于标准的 BPR 方法采样构建的三元组无法考虑到用户未浏览或购买过物品的情况,所采用的是随机采样的方法.也就是说,在随机采样过程中,可能将未来用户会购买的物品采样成为负样本,这样就潜在地降低了预测的准确度.那么如果能够确定用户购买物品的潜在规律,对随机采样施加一定的约束,避免前述将用户未来购买的物品误采样为负样本的问题,就可使训练数据更加符合用户真实的偏好,从而进一步提高排序结果的准确度.如果从社交网络中确定用户购买的规律,就可构建更加准确的用户物品三元组.

社交网络反映了用户之间的好友关系,通过与其他用户的直接连接,用户之间形成了一个网络,从而产生了更加复杂的好友关系.然后从社交网络中提取得到用户的直接好友关系和间接好友关系.从用户的直接好友关系中,进一步挖掘用户的行为和偏好,更加精准地预测用户下一步的决策.因为用户的偏好往往和其好友具有相似性,也即用户更倾向选择其好友购买或偏爱的物品.将用户之间这种偏好的相似性应用到 BPR 模型的采样中来,通过给采样过程更加合理的约束,得到更加符合用户行为的三元组,从而提高后续模型训练和推荐的准确度.

根据用户的购买或浏览记录以及社交网络,对于每个用户 u ,将该用户购买或浏览过的物品定义为 i ,将用户从未接触过的物品定义为 j ,再将该用户好友购买过的物品定义为 p .系统中所有的物品集合定义为 D ,那么用户 u 购买和浏览过的物品集合定义为 D_u ,用户好友购买过的物品定义为 D_p .最能代表用户偏好的物品首先是用户已经购买的物品 D_i ;其次,根据好友偏好的相似性,用户很可能会购买其好友购买过但是该用户未购买的物品 $D_p \setminus D_u$;最后,用户最不可能购买的物品即为 $D \setminus (D_u \cup D_p)$.根据社交网络信息构建用户和物品的三元组作为训练集,该训练集 T 可以被表示为

$$T := \{(u, i, j) | i \in (D_u \cup D_p), j \in D \setminus (D_u \cup D_p)\} \quad (5)$$

其中: (u, i, j) 为用户物品三元组,代表用户 u 对于物品 i 的偏好程度大于物品 j .其中物品 i 属于用户购买过或者用户直接好友购买过的物品,物品 j 属于用户未购买过且其直接好友也未购买过的物品.这样就基于用户的直接好友关系构建了用户物品三元组,用于后续 BPR 模型的训练.然后使用 BPR 模型构建了基于神经网络的整体推荐框架,采用 pairwise 的学习训练整个神经网络即可得到用户和物品的特征向量,从而预测用户偏好的物品,并给出这些物品的排序.

提出融合文本和评分数据进行用户和物品特征学习的方法.在前面已经构建了基于 BPR 的用户和物品的神经网络框架用来学习用户和物品的排序结果,进一步地,需要通过用户和物品之间的交互信息构建用户和物品的特征向量.假设存在一个融合特征的函数 $f(\cdot)$,从多种数据中学习到的特征表示为 \mathbf{x}_1 、 \mathbf{x}_2 和 \mathbf{x}_3 ,那么通过融合函数则可以得到融合后的特征:

$$\mathbf{x} = f(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) \quad (6)$$

其中 \mathbf{x} 为融合后的特征.这里用户和物品都由评分和评论数据的特征融合得到,所以基于该函数就可以得到用户和物品的融合特征.那么如何确定特征融合函数 $f(\cdot)$ 使得得到的融合特征更加准确就成为需要研究的问题.常用的融合方法包括串联、平均、相乘等,采用串联的方法将特征进行融合.假设学习得到的用户评论特征为 \mathbf{u}_1 ,得到的评分特征为 \mathbf{u}_2 ,那么通过函数 $f(\cdot)$ 得到的特征为 $[\mathbf{u}_1^T \quad \mathbf{u}_2^T]^T$.由于深度学习模型能够自主提取有效特征,所以串联得到了更加有效的特征表示,且不会使得原有特征由于融合而损失.使用简单的串联方式进行融合,能够增强用户和物品特征的扩展性,这对于基于多源异构数据的模型具有重要意义.随着网络中多源异构数据的增加,挖掘和提取有效信息越来越困难,而对于不同的数据需要采用不同的挖掘算法.为了减少人工的参与,更大程度增强模型的适应性,采取串联的特征融合,这样便于引入其他数据特征.假如添加一种新的异构数据,那么只需要选择一种基于深度学习的算法,并将其得到的特征加入到融合特征中,即可实现对该数据的模型扩展.

最后,提出统一的目标函数对模型进行优化.前文已经提出多源异构数据的融合函数,还需要根据融合特征构建目标函数,使得学习过程中融合特征能够更加准确代表用户或物品的特征.输入目标

函数的由用户物品三元组 (u, i, j) 所代表的对应用户的物品表示, \mathbf{u} 表示用户 u 的融合特征向量, \mathbf{i} 表示物品 i 的融合特征向量, 同理 \mathbf{j} 表示物品 j 的融合特征向量. 根据前面的定义可知用户 u 对物品 i 的偏好程度大于物品 j . 使用函数 $g(\cdot)$ 来表示结合用户和物品特征表示的损失函数, 此处将 $g(\cdot)$ 定义为 sigmoid 函数来计算用户对不同物品的不同偏好程度, 那么有 $g(\mathbf{u}, \mathbf{i}, \mathbf{j}) = \sigma(\mathbf{u}^T \mathbf{i} - \mathbf{u}^T \mathbf{j})$. 所以整个融合多源异构数据推荐模型的目标函数见式(7), L_1 和 L_2 分别为评论目标函数和评分目标函数.

$$\begin{aligned} \max_{W, \Theta} L = & \sum_{u, i, j} g(\mathbf{u}, \mathbf{i}, \mathbf{j}) + \lambda_1 L_1 - \lambda_2 L_2 = \\ & \sum_{u, i, j} \{ \sigma(\mathbf{u}^T \mathbf{i} - \mathbf{u}^T \mathbf{j}) + \\ & \lambda_1 \left(\sum_{w \in V(u, v) \in R} \sum_{d_{uw} \in R} f_{w, d_{uw}} \log \sigma(\mathbf{w}^T \mathbf{d}_{uw}) + \right. \\ & \left. \sum_{w \in V(u, v) \in R} \sum_{d_{uw} \in R} f_{w, d_{uw}} (t E_{w_N \sim P_V} \log \sigma(-\mathbf{w}_N^T \mathbf{d}_{uw})) \right) - \\ & \left. \lambda_2 (\phi(U_2 \phi(U_1(\mathbf{r}_u \odot \mathbf{r}_i) + c_1) + c_2) - r_{ui})^2 \right\} \quad (7) \end{aligned}$$

其中: W 为每种模型的权重参数, 在评论表示学习模型中用户的每条评论的权重参数都不相同, 需要通过学习得到, 而在评分计算模型中, 求得的直接是用户和物品的特征, 也就是权重参数设置为 1 即可, 不需要通过优化目标函数进行更新; $f_{w, d_{uw}}$ 代表单词和文档对出现的频率; $E_{w_N \sim P_V}$ 代表在噪声分布 P_V 上的期望值; Θ 代表模型中需要学习的其他参数, $\Theta = \{\Theta_1, \Theta_2\} = \{\{\mathbf{w}, \mathbf{d}_{uw}\}, \{U_1, U_2, c_1, c_2, \mathbf{r}_u, \mathbf{r}_i\}\}$; λ 为每种模型的惩罚参数, 其取值均在 $[0, 1]$ 区间上. 评分模型的目标函数之前添加了负号, 因为评分模型的目标函数需要最小化, 而总体模型的目标函数是最大化. 需要说明的是, 每种模型的参数 Θ 都是相互独立的, 其更新与其他方法无关. 比如评论的表示学习模型只与评论特征有关, 对应计算出用户和物品的评论特征向量. 评分特征学习模型只与评分有关, 对应计算出基于评分的用户和物品特征表示. 而整体的目标函数中, $g(\mathbf{u}, \mathbf{i}, \mathbf{j})$ 和评论、评分特征均产生关联, 所以在整体目标函数中, 所有参数均得到优化, 最后得到更精确的用户和物品的特征表示.

式(7)可以使用随机梯度下降法进行求解, 现有的深度学习框架均已集成随机梯度下降算法, 可调用方法库的随机梯度下降算法获取最终的用户和物品特征向量. 最后, 个性化的推荐列表通过用户和物品的特征向量相乘得到

$$s = \mathbf{u}^T \mathbf{v} \quad (8)$$

如式(8)所示, 每个用户和其他未购买物品的特征向量相乘即可得到该用户对该物品的购买概率, s 越大代表用户越有可能购买物品. 将所有物品的得分降序排列取前 N 个就可求得用户的 Top-N 推荐列表.

2.2 推荐模型的 Tensorflow 实现

Tensorflow 是一个开源的机器学习框架. Tensorflow 非常灵活, 基于 Tensorflow 可实现大量需要训练或推理的算法, 包括神经网络和其他机器学习算法. Tensorflow 框架被广泛应用于语音识别、计算机视觉、机器人、信息检索和自然语言处理等诸多领域.

本模型使用 Tensorflow 作为深度学习模型的实现框架, 数据处理使用的 python 库包括 numpy 和 pandas. 模型构建使用 sigmoid 交叉熵计算函数 sigmoid_cross_entropy_with_logits 计算模型的损失. 训练过程使用随机梯度下降函数 GradientDescentOptimizer 更新模型参数. 训练过程中的损失值可在 Tensorboard 中查看, 图 3 所示为损失的变化.

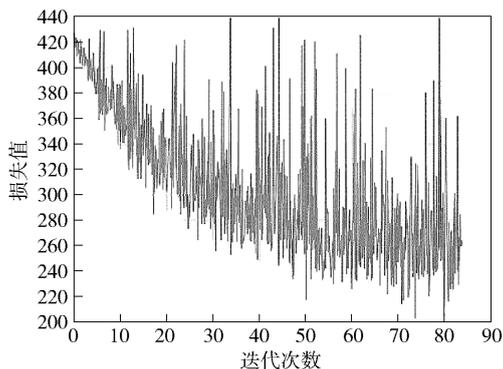


图3 Tensorboard 中的损失变化

Tensorflow 的计算流程设计为有向图形式, 图中包含了多个计算节点. Tensorflow 的图代表了数据流的计算, 图中节点具有不同分工, 部分节点用来储存或者更新持久化的状态, 部分节点用来控制分支或者循环. 在 Tensorflow 图中, 每个节点可能有零个或者更多的输入, 同样也可能有零个或者更多的输出, 每个节点代表一个操作的实例. 所提出的推荐模型计算图如图 4 所示, 展示了模型的计算节点和节点之间的数据流动. 图的数据输入为 user_id、review_id、word_id 和 product_id, 其他参数在变量节点中进行初始化. 节点 build_emb 根据数据类型和种类的不同构建特征向量, 构建完成后数据进入梯度

节点,该节点包含各类数据的损失函数.将参数传入 `clip_norm` 中修正梯度值,防止参数值更新过快.然后由更新器节点结合学习率 `le_rate` 执行随机梯度下降优化器进行反向传播.更新的参数再次传入 `build_emb`,这样就可完成一次特征向量的更新.更新过程中由存储器节点进行参数保存,为下一次训练做好准备.训练完成后,使用 `user_id` 输入 `pre_score` 节点预测用户前往商家的概率并进行排序,即可得到推荐列表.

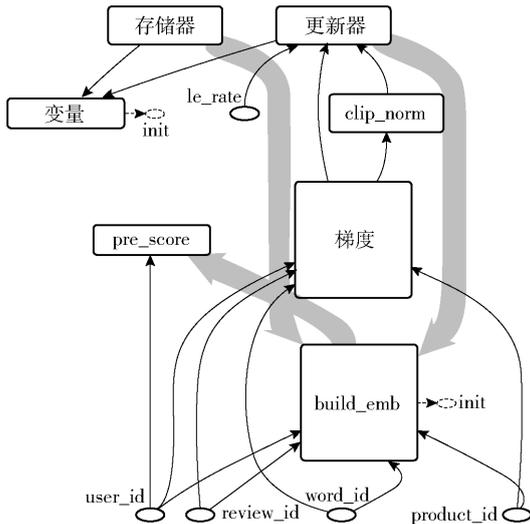


图 4 多源异构数据推荐模型计算图

3 实验

3.1 数据集

实验使用 Yelp 公开数据集. Yelp 公布的 2016 年数据集包含 686 556 位用户、85 539 位商家和 2 685 066 条评论. Yelp 数据集使用 JSON 格式存储,包含用户、商家的详细信息.数据包含用户和商家的 ID、用户对商家的评论和评分、用户间的好友关系.用户之间的社交关系被处理成用户好友关系对.评论和评分信息被用来分析用户和商家的偏好和特征.

由于数据集过于稀疏,需过滤掉部分评论较少的用户,以验证所提出模型的有效性.从 Yelp 数据集中抽取了用户评论数大于 20 的数据,此数据集被命名为 SubYelp. SubYelp 数据集的统计数据如表 1 所示.

3.2 对比试验

实验 1 的评价方法使用 Recall、Precision、归一化折损累计增益 (NDCG, normalized discounted

表 1 SubYelp 数据集的详细统计数据

数据集	SubYelp
过滤条件	用户评论数 > 20
用户	17 236
商家	72 335
社交关系	169 222
评论数	917 462
稀疏度	99.93%

cumulative gain) 和 Hit. Recall 为召回率,用来衡量算法推荐的物品中用户购买过的物品占用户所有购买过物品的比例. Precision 为准确率,用户来计算给用户推荐的物品中用户购买过的物品比例. NDCG 是用户来计算推荐结果的排名质量. Hit 为命中率,指的是推荐的物品是否有用户购买过的物品,若有则命中,反之未命中.实验 2 的评价方法使用了 Recall、Precision、NDCG.表现最好的指标用黑体标出.

实验 1 的主要目的是比较添加社交网络对算法各项指标的影响,其次比较了不同文本处理算法对实验准确度的影响.

实验 1 对比的算法如下.

BPR + Text (BT): 基于 BPR 框架,使用 PV-DBOW 算法学习评论文本特征,为用户推荐物品.

BPR + Text + Social (BTS): 基于 BPR 框架,使用评论文本和社交网络信息进行推荐.评论文本的特征表示使用 PV-DBOW 算法学习,社交网络添加在采样过程中.

HDC + Text + Social (HTS): 采用基于上下文的正则化词嵌入语言模型学习评论文本的特征表示,同时结合社交网络进行推荐.

SEL + Text + Social (STS): 采用简化的词嵌入语言模型学习评论文本的特征表示,同时结合社交网络进行推荐.

BPR + Text + Rate (BTR): 基于 BPR 框架,结合评论文本信息和评分信息进行推荐.

BPR + Text + Rate + Social (BTRS): 笔者提出的模型.基于 BPR 框架,结合评论文本信息、评分信息和社交网络信息进行推荐.

实验 1 首先对比添加社交网络前后各指标的变化.添加社交网络后,BTS 模型的准确度要高于 BT 模型;然后尝试使用不同的文本处理方法,经过比较,BTS 的准确度显著高于 HTS 和 STS 模型;最

后,实验计算了结合文本和评分的 BTR 模型及添加社交网络之后的 BTRS 模型,实验结果表明,BTRS 模型的各项指标均优于实验一所对比的其他模型. 实验结果见表 2.

实验 2 比较了笔者提出的推荐模型和其他算法.

实验 2 对比的算法如下.

BPR: 基于评分数据的贝叶斯个性化推荐模型^[10].

LDA(Latent Dirichlet allocation): 基于评分数据的一种采用吉布斯采样的狄利克雷分布生成模型^[11].

RankALS(ranking alternating least square): 基于评分数据的一种基于交替最小二乘法的个性化推荐

模型^[12].

JRL(joint representation learning): 基于评论文本和评分数据,使用简化词嵌入语义模型学习文本特征,然后使用神经网络结合文本和评分特征进行推荐的模型^[13].

BTRS(BPR + Text + Rate + Social): 提出的模型. 基于 BPR 框架,结合评论文本信息、评分信息和社交网络信息进行推荐.

BPR、LDA 和 RankALS 均为基于评分的推荐算法,JRL 是结合了评分和评论的深度学习方法,提出的 BTRS 是基于评分、评论、社交网络信息的推荐模型. 实验结果表明,对比准确率、召回率和 NDCG 指标,笔者提出的基于深度学习的融合多源异构数据的混合推荐模型具有最高准确度. 实验结果见表 3.

表 2 基于 SubYelp 的不同评价指标实验结果对比

数据集	评价指标/%	算法名称					
		BT	BTS	HTS	STS	BTR	BTRS
SubYelp	Recall@ 10	2. 826	2. 996	1. 0927	1. 059	3. 093	3. 486
	Recall@ 5	1. 697	1. 838	0. 672	0. 652	1. 881	2. 275
	Precision@ 10	3. 179	3. 362	1. 165	1. 22	3. 408	3. 809
	Precision@ 5	3. 494	3. 736	1. 315	1. 36	3. 885	4. 369
	NDCG@ 10	3. 756	4. 052	1. 394	1. 446	4. 115	4. 630
	NDCG@ 5	3. 664	3. 99	1. 3601	1. 425	4. 075	4. 770
	Hit@ 10	26. 687	27. 342	9. 806	10. 201	28. 433	30. 871
	Hit@ 5	17. 791	18. 685	6. 632	6. 847	19. 41	23. 145

表 3 基于 SubYelp 的不同评价指标实验结果对比

数据集	评价指标/%	算法名称				
		BPR	LDA	RankALS	JRL	BTRS
SubYelp	Recall@ 10	2. 192	1. 173	3. 729	3. 093	3. 486
	Recall@ 5	1. 217	0. 711	2. 176	1. 881	2. 275
	Precision@ 10	1. 932	0. 975	3. 31	3. 408	3. 809
	Precision@ 5	2. 124	1. 139	3. 839	3. 885	4. 369
	NDCG@ 10	2. 09	1. 111	3. 648	4. 115	4. 630

关于参数设置,实验 1 中深度模型的 batch size 设置为 64,最大迭代次数设置为 40,学习率设置为 0. 05,负采样数设置为 5,梯度修剪参数设置为 5,每个特征向量的维度设置为 300. 实验 2 中 BPR 的因子维度数设置为 100,最大迭代次数设置为 40,学习率设置为 0. 01,惩罚参数设置为 0. 1. LDA 的因子

维度数设置为 100,最大迭代次数设置为 3 000. RankALS 的因子维度数设置为 10,最大迭代次数设置为 5. JRL 的 batch size 设置为 64,最大迭代次数设置为 30,学习率设置为 0. 05,负采样数设置为 5,梯度修剪参数设置为 5,每个特征向量的维度设置为 200. BTRS 的设置同实验 1.

4 结束语

提出的基于深度学习的推荐模型融合评分、评论和社交网络数据进行推荐,模型采用基于深度学习的文本段落表示学习方法,设计了基于评分学习用户和物品特征的神经网络,并通过社交网络约束 pairwise 的学习. 模型弹性,可扩展,可方便融合新的异构数据,实验结果证明了模型显著提高了推荐准确度.

参考文献:

- [1] Ji Zhenyan, Yao Weina, Wei Wei, et al. Deep multi-level semantic hashing for cross-modal retrieval [J]. IEEE Access, 2019, 7: 23667-23674.
- [2] Dziugaite G K, Roy D M. Neural network matrix factorization [J]. Computer Science, 2015, arXiv: 1511.06443.
- [3] He Xiangnan, Liao Lizi, Zhang Hanwang, et al. Neural collaborative filtering[C]//Proceedings of the 26th International Conference on World Wide Web-WWW*17. New York: ACM Press, 2017: 173-182.
- [4] Li Sheng, Jaya Kawale, Fu Yun. Deep collaborative filtering via marginalized denoising auto-encoder [C] // CIKM. New York: ACM Press, 2015: 811-820.
- [5] Zheng Yin, Tang Bangsheng, Ding Wenkui, et al. A neural autoregressive approach to collaborative filtering [C]//ICML. New York: ACM Press, 2016: 764-773.
- [6] Elkahky A M, Song Yanag, He Xiaodong. A multi-view deep learning approach for cross domain user modeling in recommendation systems[C]//Proceedings of the 24th International Conference on World Wide Web-WWW*15. New York: ACM Press, 2015: 278-288.
- [7] Zheng Lei, Noroozi Vahid, Yu Philip. Joint deep modeling of users and items using reviews for recommendation [C]//Proceedings of the 10th ACM International Conference on Web Search and Data Mining-WSDM*17. New York: ACM Press, 2017: 425-434.
- [8] Shen Xiaoxuan, Yi Baolin, Zhang Zhaoli, et al. Automatic recommendation technology for learning resources with convolutional neural network[C]//2016 International Symposium on Educational Technology (ISET). New York: IEEE Press, 2016: 30-34.
- [9] Ji Zhenyan, Pi Huaiyu, Wei Wei, et al. Recommendation based on review texts and social communities: a hybrid model [J]. IEEE Access, 2019, 7: 40416-40427.
- [10] Rendle Steffen, Freudenthaler Christoph, Gantner Zeno, et al. BPR: Bayesian personalized ranking from implicit feedback[C]//Conference on Uncertainty in Artificial Intelligence. Arlington, Virginia: AUAI Press, 2009: 452-461.
- [11] Tom Griffiths, Gibbs sampling in the generative model of latent dirichlet allocation [EB/OL]. 2002 [2019-09-16]. <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.138.3760>.
- [12] Takács, Gábor, Tikk D. Alternating least squares for personalized ranking[C]//Proceedings of the 6th ACM conference on Recommender systems RecSys*12. New York: ACM Press, 2012: 83-90.
- [13] Zhang Yongfeng, Ai Qingyao, Chen Xu, et al. Joint representation learning for top-n recommendation with heterogenous information sources [C] // Proceedings of the 2017 ACM Conference on Information and Knowledge Management-CIKM*17. New York: ACM Press, 2017: 1449-1458.