

基于多源特征融合的推荐算法^①

白露露, 张 娜

(河南财政金融学院 计算机与人工智能学院, 郑州 450046)

通信作者: 张 娜, E-mail: zhangna@hafu.edu.cn



摘 要: 传统推荐算法仅依靠用户的历史评分数据来挖掘相似用户, 这一方法会面临着严重的数据稀疏性和冷启动问题, 造成推荐效果不理想. 为解决数据稀疏性, 提高推荐质量, 通过深度挖掘评论文本和评分数据的语义信息, 提出了基于多源特征融合的推荐算法模型 (recommendation algorithm model based on multi-source feature fusion, MSFF), 该模型通过 CNN 和 SA_BiLSTM 混合模型对评论文本信息进行特征提取, 并在特征提取时引入自注意力机制动态地调整评论信息的重要性, 从而更好地捕捉评论信息中的深层关键信息. 接着通过隐语义模型 LFM 分解评分矩阵得到用户和物品潜在特征. 然后将两种模型所得的非同源特征进行有效融合得到高阶特征, 进而预测评分, 并根据最终的预测评分生成推荐列表. 为了验证模型的有效性, 在 Yelp 评论数据集和亚马逊评论数据集 Amazon5-core 中的 3 个子数据集上进行实验, 结果表明, 该模型不仅缓解推荐算法中评分数据稀疏性的影响, 而且有效地提升了推荐结果的准确性.

关键词: 推荐算法; 评论文本; 评分预测; 多源特征融合; 矩阵分解

引用格式: 白露露, 张娜. 基于多源特征融合的推荐算法. 计算机系统应用, 2025, 34(11): 289-299. <http://www.c-s-a.org.cn/1003-3254/10024.html>

Recommendation Algorithm Based on Multi-source Feature Fusion

BAI Lu-Lu, ZHANG Na

(School of Computer and Artificial Intelligence, Henan Finance University, Zhengzhou 450046, China)

Abstract: Traditional recommendation algorithms rely solely on users' historical rating data to identify similar users, which will suffer from severe data sparsity and cold start problems and thus result in unsatisfactory recommendation performance. A recommendation algorithm model based on multi-source feature fusion (MSFF) is proposed to address data sparsity and improve recommendation quality, which deeply mines the semantic information in review texts and rating data. This model employs a hybrid model of CNN and SA_BiLSTM to extract features from review texts, and introduces a self-attention mechanism during feature extraction to dynamically adjust the importance of review information, thereby better capturing the deep key information in the review information. Subsequently, the latent factor model (LFM) is adopted to decompose the rating matrix to obtain latent features of users and items. Then, the non-homogeneous features obtained from the two models are effectively fused to generate high-order features, which are adopted to predict ratings. Finally, a recommendation list is generated based on the predicted ratings. Experiments are conducted on the Yelp review dataset and three sub-datasets in the Amazon review dataset, Amazon5-core, to validate the effectiveness of the model. The results demonstrate that the model not only mitigates the influence of rating data sparsity in recommendation algorithms but also improves the accuracy of recommendation results.

Key words: recommendation algorithm; comment text; score prediction; multi-source feature fusion; matrix factorization

① 基金项目: 河南省科技攻关计划 (212102210501); 河南省高等学校重点科研项目 (20A520008)

收稿时间: 2025-03-05; 修改时间: 2025-05-07; 采用时间: 2025-07-07; csa 在线出版时间: 2025-09-30

CNKI 网络首发时间: 2025-10-09

近年来,随着互联网的快速发展,网上的信息呈现指数级增长,用户在海量的信息资源中无法快速、准确地找到想要的信息.为了帮助用户解决选择困难问题,摆脱信息过载带来的困扰,推荐系统应运而生^[1].推荐算法通过用户历史行为来推断用户兴趣爱好,并为其推荐感兴趣的物品和信息,有效地改善信息过载问题^[2].

传统的推荐算法主要是基于协同过滤算法,该算法通过用户的历史评分数据对其兴趣爱好进行分析,然后为其推荐喜欢的物品.但是评分数据往往存在稀疏性问题,很难训练出精准的推荐模型,影响推荐的准确度,另外,仅依靠评分数据也很难完整地解释用户的兴趣偏好和物品的属性特征^[3].因此为了解决评分数据稀疏性问题,许多学者选择通过添加各种辅助信息提升推荐准确度,例如:评论文本、社交网络^[4,5]、图结构^[6,7]等信息.其中评论文本不仅与用户评分紧密相关,可以弥补评分稀疏性的问题^[8].Zhang等人^[9]提出了一种结合多模信息和注意力机制的推荐算法,用CNN和RNN分别提取图像和文本中的特征后再进行组合,然后用注意力机制来模拟标签和其余信息的关联,从而实现推荐,有效解决了数据稀疏性问题.Chen等人^[10]利用改进的双向GRU分别对用户的评论信息和商品的评论信息进行特征提取,预测用户对商品的评分.Lu等人^[11]提出一种基于注意力的GRU网络,从评分和用户评论中共同学习用户和项目特征信息,进行评分预测.Chen等人^[12]依据评论信息进行建模,并利用注意力机制来区分不同评论的重要程度,相比依据基础评论文档建模的模型推荐准确度得到很大提升.

以上模型虽然有效地缓解了数据稀疏性问题,但是在特征提取时模型比较单一,忽略了数据间的高阶性,造成推荐的准确度不够理想.而且单一的特征提取模型性能也是有限的,基于此,本文提出一种基于多源特征融合的推荐算法模型,该模型充分利用评论信息和评分矩阵中的隐性信息,对非同源特征进行融合,不但克服评分矩阵的稀疏性,而且更加有效地提升推荐的准确性,帮助用户得到所感兴趣的物品.

1 相关工作

早期大部分推荐算法是基于用户-物品历史评分矩阵进行搭建^[13,14].其中最为经典的是协同过滤算法^[15,16],该算法的基本原理是根据用户的行为和偏好,发现用户之间的相似性或者物品之间的相似性,进而

为用户推荐他们可能感兴趣的物品,但是对于一些新用户或者新物品,由于缺乏历史数据,容易导致推荐的冷启动问题.为了缓解冷启动问题,Zhang等人^[17]将矩阵分解技术引入到协同过滤算法中,通过将用户偏好预测评分.虽然一定程度上解决了协同过滤面临的冷启动问题,但是矩阵分解技术对评分矩阵的稀疏性依赖性很高,这就导致在数据稀疏性过高的情况下,不能准确地提取到用户偏好的物品的属性信息,造成推荐性能下降.为了克服该问题,文献^[18,19]将用户的社交信息引入推荐算法中,通过分析好友的喜好来预测用户的喜好,从而形成推荐.Kim等人^[20]利用物品评论信息进行建模,即利用卷积神经网络对商品评论文本中商品特征进行提取,然后与矩阵分解技术相结合和物品属性表示成潜在隐向量,然后通过内积计算得到实现评分预测.Serra等人^[21]提出了TLCM-CNN模型对每个用户所有评论的词频进行编码,从全局文本中学习用户和商品的潜在特征,然后对评分进行预测.Suryana等人^[22]为了考虑评论文本信息中的上下文信息,利用LSTM模型进行文本特征提取,既克服了数据稀疏性问题,又提高了推荐准确度.Shuai等人^[23]构建了一个以评论为基础的评论感知用户-项目图模型,以辅助用户和项目的表示学习.Liu等人^[24]为了更好地捕捉用户的兴趣,在利用评论文本进行特征提取时都引入注意力机制,为文本中不同词赋予不同的权重对用户和物品建模.

本文在前面学者的研究基础上,将深度学习技术CNN和SA_BiLSTM模型进行融合对评论信息进行特征提取,然后融入评分矩阵,预测用户与商品的评分.该模型首先将用户和商品的所有评论连接起来形成两个长的文档,使用词嵌入模型Word2Vec将其转为词向量,分别使用CNN和SA_BiLSTM提取文本的局部特征和全局特征,同时在提取全局特征时引入自注意力机制,捕捉评论信息中的关键信息.接着将提取到的特征信息进行拼接融合并融入评分矩阵得到高阶特征,从而得到最终的评分,形成推荐列表.

2 基于多源特征融合的推荐算法模型

基于多源特征融合的推荐算法如图1所示,该模型主要由4个模块组成.

模块1:文本特征提取模块,主要使用CNN和SA_BiLSTM混合网络模型分别对用户评论文本和物品评

论文本进行建模,得到用户和物品的深层潜在特征。

模块 2: 基于 LFM 的用户和物品潜在特征生成模块,主要是利用隐语义模型对评分矩阵进行分解,得到用户和商品的隐特征向量 P 和 Q 。

模块 3: 特征融合模块,将文本特征处理模块和基

于 LFM 模型得到的非同源用户和项目特征进行融合,然后利用因子分解机和多层感知机 MLP 对融合后的特征进行处理,得到高级的用户和项目特征。

模块 4: 评分预测模块,主要将得到的用户高级特征和商品高级特征输入全连接层,得到最终的评分 R 。

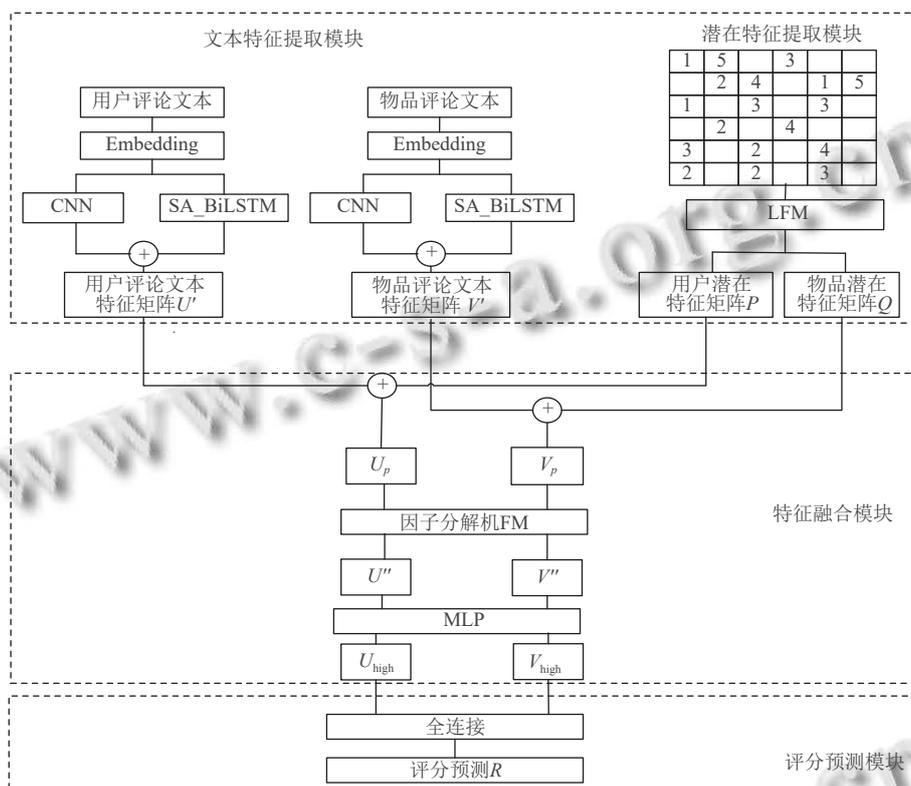


图 1 MSFF 模型结构

2.1 文本表示

在文本特征提取中,需要将文本转换成计算机能识别或计算的向量形式,即将文本中的每个单词或短语转换成一个固定长度的密集向量。本文使用词嵌入(Word2Vec)模型中的 Skip-Gram 方法来获得评论文本的语义信息。相比其他词嵌入模型, Skip-Gram 模型训练时间短且效果好, Skip-Gram 模型在进行 Embedding 训练时,根据当前词来预测上下文的词,将词映射到一个低维空间,并且低维空间中的词向量的位置关系可以很好地表示其在语义上的联系,进而更好地反映文本的特征。假设在评论文本中,用户 u 对物品 i 的一条评论 $S_{u,i} = (S_1, S_2, S_3, \dots, S_n)$ 包含 n 条语句 S , 其中一条语句 $S = (s_1, s_2, s_3, \dots, s_k)$ 含有 k 个单词,使用词嵌入矩阵 $E \in R^{V \times D}$ 将语句转化为词向量矩阵 $(w_1, w_2, w_3, \dots, w_k)$, 其中 V 和 D 分别表示词汇量和词嵌入维

度, w_k 为语句中单词对应的词嵌入向量。

2.2 文本局部特征提取

由于每个用户可能会购买多个物品,并产生多条评论信息,而这些信息可以更好地反映用户的偏好。同时一个物品又可能被多个用户评论,物品的评论更多的是包含物品的属性信息。因此,本文将单个用户所有的评论进行连接,形成一个较长用户评论文本 $D_u = D_{u1}, D_{u2}, D_{u3}, \dots, D_{um}$, 将单个物品的所有评论信息进行连接,形成一个较长物品评论文本 $D_v = D_{v1}, D_{v2}, D_{v3}, \dots, D_{vn}$ 。然后对用户评论文本 D_u 和物品评论文本 D_v 进行特征提取得到用户和物品特征。由于 CNN 可以通过卷积层有效捕捉文本序列中的局部特征,而 SA_BiLSTM 可以学习文本序列中的长期依赖关系,捕捉全局信息,将两种模型进行组合,可以同时关注文本序列的局部和全局信息,提高模型对序列中复杂模式的识别能力。

本文选择 CNN 网络对用户评论信息进行局部特征提取, CNN 网络主要由卷积层和池化层两部分构成, 其工作原理如图 2 所示。

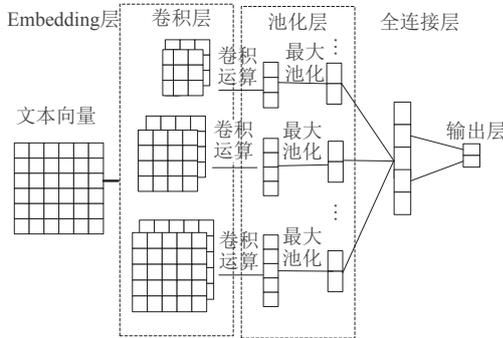


图 2 CNN 模型结构

- 卷积层主要通过不同尺寸大小的卷积核对文本进行卷积操作, 每个卷积核按照一定的步长对输入的文本矩阵进行扫描, 通过计算被扫描的当前区域与卷积核之间的点积, 生成特征矩阵。由于不同尺寸的卷积核表示不同大小的感受野, 因此本文通过多次实验最终选择卷积核大小为[3, 4, 5]来进行卷积操作提取不同层次的特征信息, 然后将多层信息进行拼接组合, 作为卷积操作的输出, 计算结果如式 (1):

$$m_i = f(w \times x_{i:i+h-1} + b) \quad (1)$$

其中, m_i 表示第 i 个卷积核提取的文本特征, f 为 ReLU 激活函数, w 为卷积核权重参数, b 为偏置信息, x_i 为 x 的第 i 个特征。

将上述所有特征连接起来得到卷积层数输出特征 M , 如式 (2):

$$M = [m_1, m_2, m_3, \dots, m_{i+h-1}] \quad (2)$$

- 池化层是对卷积层输出进行空间降维, 也是对特征进一步筛选提取出更加显著的特征, 同时保持特征的不变性。常见的池化有最大池化 (max pooling) 和平均池化 (average pooling), 本文选择最大池化对特征矩阵进行池化操作, 即提取特征矩阵 M 中最为突出的一个特征代替所有特征, 具体操作如式 (3):

$$m_j = \max[m_1, m_2, m_3, \dots, m_{i+h-1}] \quad (3)$$

其中, 池化层使用若干个 $n \times m$ 的滤波器依次进行池化操作, 将池化结果连接起来构成池化结果表示为 $u_c = [u_1^c, u_2^c, u_3^c, \dots, u_n^c]$ 。

同理, 使用 CNN 对物品的评论信息进行局部特征提取得到物品的特征矩阵信息可以表示为 $v_c = [v_1^c, v_2^c, v_3^c, \dots, v_n^c]$ 。

2.3 文本全局特征提取

CNN 虽然在文本特征提取中, 取得了较好的结果, 但是很难把握序列关系以及上下文信息, 因此本文引入 BiLSTM 弥补 CNN 在保留和控制历史信息能力上的不足, 从而更好地捕捉评论信息文本的上下文语义信息。BiLSTM 是 LSTM 网络的变体, 其隐藏层由前向 LSTM 与后向 LSTM 共同构成, 通过双向对文本进行编码, 提取上下文语义特征并且捕捉长距离依赖关系。LSTM 模型主要包括 3 个不同的门控制单元和 1 个记忆单元, 3 个门分别是“输入门”“遗忘门”和“输出门”, 其在时间步 t 时刻的状态转移公式如式 (4)–(9):

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (4)$$

$$\tilde{C}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \quad (5)$$

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (6)$$

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \quad (7)$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (8)$$

$$h_t = o_t \cdot \tanh(c_t) \quad (9)$$

其中, i_t 输入门的输出, 表示哪些信息被更新, f_t 遗忘门的输出, 表示哪些信息被保留, o_t 输出门的输出, 表示 t 时刻模型的输出, C_t 表示当前时刻的记忆单元状态, x_t 表示当前时刻的输入, h_{t-1} 表示上一时刻隐藏状态的输入信息, W 表示权重信息, σ 表示激活函数, b 表示偏置项。

利用 BiLSTM 对评论文本信息上下文信息提取时, 在 t 时刻的隐藏状态 h_t 是由 t 时刻前向 LSTM 模型的隐藏状态 \vec{h}_t 和反向 LSTM 模型的隐藏状态 \overleftarrow{h}_t 共同拼接而成的, 其公式如式 (10):

$$h_t = \left[\vec{h}_t, \overleftarrow{h}_t \right] \quad (10)$$

为了更好地捕捉评论文本上下文信息, 解决评论信息中单词序列中不同单词对整个句子产生不同的影响, 本文引入了自注意力机制, 即在 BiLSTM 模型后面添加一层 self-attention, 得到模型 SA_BiLSTM 模型, 从而关注文本中的重点信息, 同时也可以快速获取有效信息, 节约资源。

因此, SA_BiLSTM 模型对用户评论信息和物品评论信息进行全局特征提取得到特征矩阵表示为 $u_l = [u_1^l, u_2^l, u_3^l, \dots, u_n^l]$, $v_l = [v_1^l, v_2^l, v_3^l, \dots, v_n^l]$ 。接下来将基于 CNN 网络模型提取的用户特征信息矩阵 u_c 和基于 SA_BiLSTM 网络模型提取的用户特征信息阵 u_l 进行连接

得到最终用户评论特征矩阵 $U' = [u_1^c + u_1^l, u_2^c + u_2^l, \dots, u_n^c + u_n^l]$, 同理得到物品最终评论特征矩阵 $V' = [v_1^c + v_1^l, v_2^c + v_2^l, \dots, v_n^c + v_n^l]$.

2.4 基于 LFM 的用户和物品潜在特征生成模块

隐语义模型 (LFM) 主要根据隐含特征将用户与商品联系起来, 利用历史评分进行迭代学习, 将用户-商品评分矩阵 R 分解成两个 K 维向量表示, 即用户潜在特征矩阵 P 和商品潜在特征矩阵 Q , 然后通过两个低维矩阵 P 和 Q 内积运算来表示用户-商品的预测评分 \tilde{R} , 具体公式如式 (11):

$$\tilde{R} = P^T Q \quad (11)$$

其中, \tilde{R} 中的每一个具体评分数据的表达式如式 (12):

$$\tilde{r}_{ui} = p_u^T q_i = \sum_{k=1}^K p_{uk} q_{ki} \quad (12)$$

为了更准确地得到用户潜在特征矩阵 P 和商品潜在特征矩阵 Q , 需要通过最小化损失函数, 损失函数的表达式如式 (13):

$$C(p, q) = \sum_{(u,i) \in \text{Train}} \left(r_{ui} - \sum_f p_{uf} q_{if} \right)^2 + \lambda (\|p_u\|^2 + \|q_i\|^2) \quad (13)$$

其中, r_{ui} 表示真实评分, $\sum_f p_{uf} q_{if}$ 表示预测评分, $\lambda (\|p_u\|^2 + \|q_i\|^2)$ 表示为防止过拟合引入的正则化项, 为使上述的损失函数最小化, 本文使用梯度下降法对参数 P 和 Q 进行更新, 最终得到最优用户潜在特征矩阵 P 和商品潜在特征矩阵 Q .

2.5 特征融合模块

由于文本特征提取模块得到的用户特征矩阵 U' 和物品特征矩阵 V' 是非线性特征向量, 而通过 LFM 模型对评分数据提取的用户潜在特征矩阵 P 和商品潜在特征矩阵 Q 是浅层的线性特征向量. 两者来自不同的数据源无法直接联合进行学习, 因此本文将两个不同数据源提取的特征进行不同层次的融合, 设计了一个结合 FM 和 MLP 的特征融合模块, 从而得到高阶特征, 完成评分预测.

(1) 非同源特征融合

如果直接将通过评论文本提取的特征向量和评分矩阵直接维度相加, 可能会忽略两种不同特征的重要性, 因此本文在对文本特征 U' 、 V' 和评分特征 P 、 Q 进行融合时, 通过权重对两种特征重要性进行调整,

从而提升融合的效率, 特征融合的计算公式如式 (14)、式 (15):

$$U_p = \mu U' + (1 - \mu) P \quad (14)$$

$$V_q = \mu V' + (1 - \mu) Q \quad (15)$$

(2) 低阶特征提取

在实际应用中, 不同特征之间是会相互影响的, 所以在特征融合时需要考虑不同特征之间的非线性关系, 进而构建组合特征, 捕捉特征之间的交互关系, 提升模型的泛化能力, 同时也可以降低特征的维度, 减少计算时间. 本文通过因子分解机模型 (factorization machine, FM) 将特征向量进行组合, 即特征两两组合形成一个新的交叉特征, FM 具体计算过程如式 (16)、式 (17):

$$\hat{y} = w_0 + \sum_{i=1}^N w_i x_i + \sum_{i=1}^N \sum_{j=i+1}^N \langle v_i, v_j \rangle x_i x_j \quad (16)$$

其中, $\sum_{i=1}^n \sum_{j=i+1}^n \langle v_i, v_j \rangle x_i x_j$ 表示交叉项, 即 FM 是在线性模型的基础上增加一个交叉项来实现特征组合, 但由于交叉项的存在, 如果直接计算会增加模型的复杂度, 所以可以通过公式变换把时间复杂度降低, 其中 k 是超参数, 为 v 的维度.

$$\begin{aligned} & \sum_{i=1}^n \sum_{j=i+1}^n \langle v_i, v_j \rangle x_i x_j \\ &= \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \langle v_i, v_j \rangle x_i x_j - \frac{1}{2} \sum_{i=1}^n \langle v_i, v_i \rangle x_i x_i \\ &= \frac{1}{2} \left(\sum_{i=1}^n \sum_{j=1}^n \sum_{f=1}^k v_{i,f} v_{j,f} x_i x_j - \sum_{i=1}^n \sum_{f=1}^k v_{i,f} v_{i,f} x_i x_i \right) \\ &= \frac{1}{2} \sum_{f=1}^k \left(\left(\sum_{i=1}^n v_{i,f} x_i \right) \left(\sum_{j=1}^n v_{j,f} x_j \right) - \sum_{i=1}^n v_{i,f}^2 x_i^2 \right) \\ &= \frac{1}{2} \sum_{f=1}^k \left(\left(\sum_{i=1}^n v_{i,f} x_i \right)^2 - \sum_{i=1}^n v_{i,f}^2 x_i^2 \right) \end{aligned} \quad (17)$$

因此将非同源特征融合得到的用户特征和项目特征再通过因子分解机模型 (FM) 得到低阶的用户交叉特征和商品交叉特征, 如式 (18)、式 (19):

$$U'' = w_0 + \sum_{i=1}^H w_i u_i^p + \frac{1}{2} \left(\left(\sum_{i=1}^H u_i^p \right)^2 - \sum_{i=1}^H (u_i^p)^2 \right) \quad (18)$$

$$V'' = w_0 + \sum_{i=1}^H w_i v_i^q + \frac{1}{2} \left(\left(\sum_{i=1}^H v_i^q \right)^2 - \sum_{i=1}^H (v_i^q)^2 \right) \quad (19)$$

其中, w_0 表示全局偏置项, w_i 表示交叉特征中的第 i 个变量的权重值。

(3) 高阶特征提取

因子分解模型的最大优点是即使数据非常稀疏对模型的影响也不是很大, 所以可以很好地解决数据的稀疏性问题。但是该模型通过参数的学习得到的是低阶特征, 无法保证特征的有效性。因此, 需要将低阶的用户特征和商品低阶特征分别送入多层感知机中, 通过不断的参数学习从而得到用户的高阶特征和项目的高阶特征, 具体公式如式 (20)、式 (21):

$$U_{\text{high}} = f(W \times U'' + b) \quad (20)$$

$$V_{\text{high}} = f(W \times V'' + b) \quad (21)$$

其中, f 表示为激活函数, W 表示为权重矩阵, b 表示为偏置项。

2.6 评分预测模块

将上述得到的高阶用户特征 U_{high} 和高阶项目特征 V_{high} 进行乘积运算, 并再次通过全连接层, 得到预测评分 \tilde{R}_{UV} , 如式 (22):

$$\tilde{R}_{UV} = w(U_{\text{high}} \otimes V_{\text{high}}) + b_u + b_v + \mu \quad (22)$$

其中, w 表示权重参数, b_u 、 b_v 分别表示用户的偏差项和商品的偏差项, μ 表示全局偏差项。计算过程见算法 1。

算法 1. 评分预测算法

输入: 用户评论文本 D_u 、商品评论文本 D_v 、用户-商品评分矩阵 R 。
输出: 用户预测评分 \tilde{R}_{UV} 。

1. 评论文本经过 Embedding 得到词向量特征;
2. 将词向量特征分别输入 CNN 和 SA_BiLSTM 模型中, 得到局部特征 u_c 、 v_c 和全局特征 u_l 、 v_l ;
3. 将局部特征和全局特征进行连接, 得到最终的用户评论特征矩阵 U' 和商品评论特征矩阵 V' ;
4. 利用 LFM 模型通过用户-商品的评分矩阵提取的用户潜在特征矩阵 P 和商品潜在特征矩阵 Q ;
5. 将评论特征矩阵和评分潜在特征矩阵进行拼接, 然后利用式 (18) 和式 (19) 进行特征融合从而得到低阶特征矩阵 U'' 和 V'' ;
6. 通过全连接层, 得到用户的高阶特征 U_{high} 和项目的高阶特征 V_{high} ;
7. 根据式 (22) 进行评分预测。

然后根据预测评分 \tilde{R}_{UV} 进行降序排序, 生成 Top- N 推荐列表给用户推荐所感兴趣的商品。

在参数优化时, 本文选用了平方差作为目标函数, 并引入正则化, 提高模型的泛化能力。最小化目标函数的表达式如式 (23):

$$\text{loss} = \min \sum_{(u,i) \in T} (\tilde{R}_{u,i} - R_{u,i})^2 + \lambda \|\omega\| \quad (23)$$

其中, T 表示用户-商品的评分集合, $\tilde{R}_{u,i}$ 、 $R_{u,i}$ 分别表示预测评分和真实评分, λ 表示正则化系数, 为了防止过拟合, 目标函数中使用 L2 正则化约束。

3 实验结果与分析

3.1 实验数据集

为了验证模型的性能, 本文使用 Yelp 评论数据集和亚马逊评论公共数据集 Amazon5-core 中的 3 个子类别数据集 musical instruments、office products、video games 作为实验数据, 每个数据集都包括“用户 ID”“商品 ID”“评分 1-5”“评论信息”这 4 个特征, 数据的基本信息统计表如表 1 所示。

表 1 数据集

数据集	用户数量	商品数量	评分数量	稀疏度 (%)
Musical instruments	1 429	900	10 261	99.20
Office products	4 905	2 420	53 228	99.55
Video games	24 303	10 672	231 780	99.98
Yelp	40 500	58 755	2 024 283	99.91

从表 1 可以看出, 虽然每一个数据集中用户和商品数目都比较大, 但是评分矩阵的密度是比较小的, 表明每个用户对商品的实际打分是比较少的, 所以评分矩阵是比较稀疏。如果只使用评分矩阵预测用户可能对商品的评分, 将严重影响预测的准确度, 因此本文利用用户的评论信息和商品评论信息作为辅助信息, 可以更好地解决推荐模型所面临的稀疏性问题。

3.2 参数设置

本实验 batch_size 批次大小为 64, 词向量的维度为 300, 卷积核个数为 128, 窗口大小为 [3, 4, 5], 学习率为 0.001, 采用 Adam 优化器更新模型参数, SA_BiLSTM 隐藏单元个数为 100, dropout 值为 0.5, LFM 模型分解的潜在特征向量维度 K 在 [5, 10, 15, 20, 25, 30] 之间进行调节, 通过实验当 $K=15$ 时, 模型性能最好。

3.3 评价指标

考虑到评分预测任务的本质是回归问题, 因此本文采用平均绝对误差 (MAE) 和均方根误差 (RMSE) 作为模型的主要评估指标, 具体公式如式 (24)、式 (25):

$$\text{MAE} = \frac{1}{T} \sum_{(u,i) \in T} |\tilde{R}_{u,i} - R_{u,i}| \quad (24)$$

$$RMSE = \frac{1}{T} \sqrt{\sum_{(u,i) \in T} (\tilde{R}_{u,i} - R_{u,i})^2} \quad (25)$$

其中, T 表示用户-商品的评分集合, $\tilde{R}_{u,i}$ 、 $R_{u,i}$ 分别表示预测评分和真实评分. 平均绝对误差 (MAE) 和均方根误差 ($RMSE$) 的值越小, 预测的准确度越高, 与 MAE 相比, $RMSE$ 增加了对错误预测的惩罚, 并且在定义上似乎更加严格.

3.4 对比模型

为了验证模型的有效性, 本文选择以下 6 种方法进行对比.

LFM^[25]模型: 隐语义模型 (latent factor model) 主要是根据用户-商品评分数据进行矩阵分解, 实现评分预测, 是目前最经典的矩阵分解算法.

DeepCoNN^[26]模型: 以两个平行的 CNN 网络分别提取用户评论和商品评论中的特征, 但局限是仅利用了评论信息解决数据的稀疏问题, 没有利用用户-商品的评分等级.

DRRNN^[27]模型: 使用 MLP 和 CNN 分别对用户评分等级和评论信息进行特征提取分别得到评分预测向

量和评论预测向量特征, 然后通过 `concat` 得到预测向量, 再输入到全连接层得到评分.

NARRE^[28]模型: 使用 BERT 得到评论嵌入, 然后将用户 ID 及物品 ID 输入多层感知机得到预测评分.

RAAM^[29]模型: 利用两个并行的 CNN 网络分别对用户和物品的文本数据进行特征提取, 并在网络中加入了三级注意力机制, 分别对单词、语句和评论的重要性进行捕捉, 提升推荐的准确度.

ACGIG^[30]模型: 提出一种结合注意力 CNN 与 GNN 的信息融合推荐方法, 利用 CNN 和 GNN 模型, 从评论、评分和信任等信息中学习用户偏好, 实现评分预测.

MSFF 模型: 本文提出的 MSFF 模型, 同时利用评分矩阵和评论文本进行特征提取得到评分交互特征和文本特征, 然后通过因子分解机进行特征融合, 从而进行评分预测.

3.5 性能评估

本文所提出的 MSFF 和其他对比模型的评分预测结果如表 2 所示.

表 2 各算法的 $RMSE$ 与 MAE 对比结果

Method	Musical instruments		Office products		Video games		Yelp	
	$RMSE$	MAE	$RMSE$	MAE	$RMSE$	MAE	$RMSE$	MAE
LFM	1.273	1.043	1.347	1.076	1.403	1.222	1.394	1.113
DeepCoNN	0.875	0.854	0.924	0.876	0.965	0.903	0.915	0.871
DRRNN	0.864	0.827	0.877	0.832	0.934	0.895	0.892	0.851
NARRE	0.856	0.835	0.865	0.821	0.897	0.862	0.886	0.8583
RAAM	0.851	0.830	0.857	0.838	0.884	0.844	0.883	0.863
ACGIG	0.840	0.825	0.837	0.829	0.865	0.848	0.874	0.856
MSFF	0.825	0.810	0.801	0.786	0.841	0.828	0.869	0.840

从表 2 可知, LFM 方法性能在 3 个数据集上表现都是最差的, 尤其当数据量比较大且比较稀疏时表现得更差. 其主要原因是 LFM 仅利用用户-项目的评分信息预测评分, 当数据比较稀疏时, 预测的准确度较低. 而当模型融入了辅助信息评论文本时, 可以很大程度上提高评分预测的准确度, 同时很好地解决数据的稀疏性问题. 而对于同样都考虑了评论信息的模型, 使用多个模型进行特征提取要比使用单个模型进行特征提取的效果要好, 如 NARRE、DRRNN 和本文提出的 MSFF 模型的 $RMSE$ 和 MAE 都要比 DeepCoNN 低, 主要因为 DeepCoNN 仅使用 CNN 分别对用户评论和项目评论进行特征提取, 而 CNN 只能学习局部特征, 对于长序列文本并不能考虑到上下文信息, 造成丢失信息过多. NARRE 模型是在 DeepCoNN 的基础上加入

了注意力机制来判断评论的贡献程度, DRRNN 使用 MLP 和 CNN 分别对用户评分等级和评论信息进行特征提取, 从而进行评分预测, 本文所提出的 MSFF 同时利用 CNN 和 SA_BiLSTM 两个模型对评分矩阵和评论文本进行特征提取, 不仅考虑局部特征, 同时又考虑到了长序列文本中的上下文信息特征提取, 同时还利用注意力机制判断评论的贡献度, 从而提升了评分预测的准确度.

3.6 消融实验

为了进一步验证 MSFF 算法模型中各个模块对模型具有正反馈作用, 本文在 Amazon5-core 的 office products 数据集上进行消融实验.

MSFF_A 模型: 舍弃 SA_BiLSTM 模型, 只保留 CNN 模型对评论文本进行局部特征提取和 LFM 模型对用

户潜在特征和项目潜在特征进行提取。

MSFF_B 模型: 舍弃 CNN 模型, 只保留 SA_BiLSTM 对评论文本进行全局特征提取和 LFM 模型对用户潜在特征和项目潜在特征进行提取。

MSFF_C 模型: 舍弃 LFM 模型, 只保留 CNN 模型对评论文本进行局部特征提取和 SA_BiLSTM 模型对评论文本进行全局特征提取。

由表 3 可知, 通过对 Amazon5-core office products 数据集的消融实验分析表明, 本文提出的 MSFF 模型通过 CNN、SA_BiLSTM 和 LFM 的多级特征融合虽然训练时间增加 8.7% (较 MSFF_C), 其时间代价源于多模块协同计算——CNN 局部特征提取、SA_BiLSTM 全局依赖建模及 LFM 特征交叉的迭代优化。但是在 RMSE 和 MAE 性能指标上 MSFF 模型明显比其他 3 个模型有较大改进, 其中 RMSE 降低了 7.3% (较 MSFF_C), MAE 降低了 7.2% (较 MSFF_C)。

表 3 各模块对模型影响及模型训练时间对比表

Algorithms	RMSE	MAE	训练时间
MSFF_A	0.972	0.903	28'43"
MSFF_B	0.925	0.884	23'17"
MSFF_C	0.863	0.838	32'50"
MSFF	0.804	0.782	35'42"

因此, 通过上述消融实验结果和分析验证了 MSFF 算法模型中的 CNN、SA_BiLSTM 和 LFM 这 3 个模块在评分预测中发挥了重要作用, 都能够有效的提升特征提取的效果。

3.7 模型参数分析

在深度学习模型中, 合适的参数能帮助模型提升预测的准确度, 因此本文在实验环境和实验数据集相同的情况下, 探究不同参数对模型性能的影响。

(1) 词向量维度

为了训练更具有表征能力的高质量词向量, 本文对 Word2Vec 模型中的词向量维度进行实验, 将训练出的不同维度的词向量输入 MSFF 模型中, 探究词向量维度对模型性能影响, 实验结果如图 3 所示。

由实验结果可知, 当词向量维度值为 300 时, 评分预测准确度最高, 所以本文将 Word2Vec 模型的词向量维度设置为 300。

(2) 潜在特征向量维度 K

LFM 通过分析用户-物品交互数据, 学习出一组潜在的特征, 这些特征在原始数据中并不直接显现, 但能够帮助我们更好地理解 and 预测用户的行为, 图 4 表示不同潜在特征向量维度 K 对模型的影响。

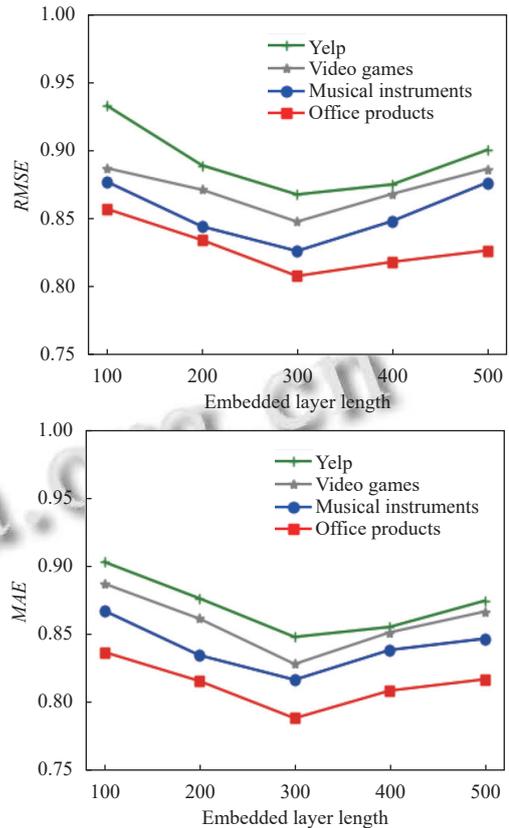


图 3 词向量维度对模型的影响

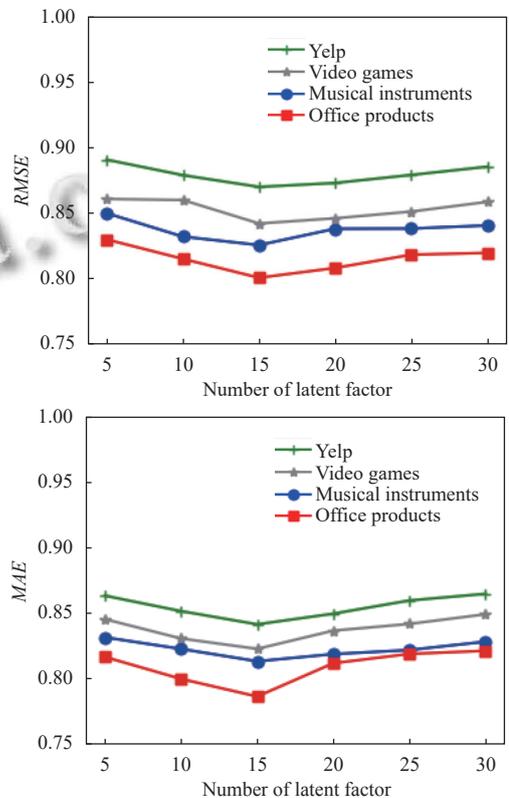


图 4 潜在特征向量维度对模型的影响

由图4可知,随着本文所提出的MSFF模型随着潜在特征向量维度 K 的增加,评分预测的性能明显得到提升,但当潜在特征向量维度 K 值由15增加到20时预测误差有一定的上升,通过分析得知,合适的潜在特征维度模型可以更好地学习用户和商品的特征,提升预测的准确度.而过大设置潜在特征维度,模型过度的学习,可能会出现过拟合,则学习的潜在特征将不能很好地表达用户和商品的真实特征.所以为了使模型性能达到最佳效果,本文将潜在特征向量维度 K 的值设置为15.

(3) Dropout

在训练网络模型时,可以选择丢弃一部分神经元来避免过拟合现象,但如果抛弃的比率选择不恰当,可能会降低模型的性能,图5表示不同的丢弃率对模型性能的影响.

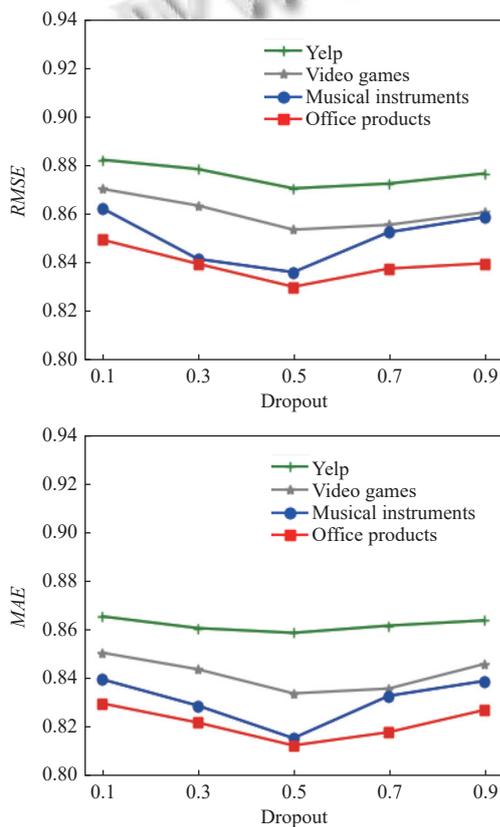


图5 Dropout对模型性能影响

由图5可知,选择合适的丢弃率有助于提升模型的性能,从而防止过拟合现象的产生,MSFF模型在4个数据集中,当dropout值设置为0.5时性能最优.但是在musical instrumets数据集的结果对丢弃率的变化

更为敏感.主要是因为该数据规模较小,导致模型在没有丢弃机制时更容易出现过拟合现象.

(4) 是否加入 Attention 机制

为了解决长文本中单词序列中不同单词对整个句子的影响,本文在双向BiLSTM模型后加入了Attention机制从而更加关注文本中单词权重比较高的单词.图6表示是否引入Attention机制对模型性能的影响.

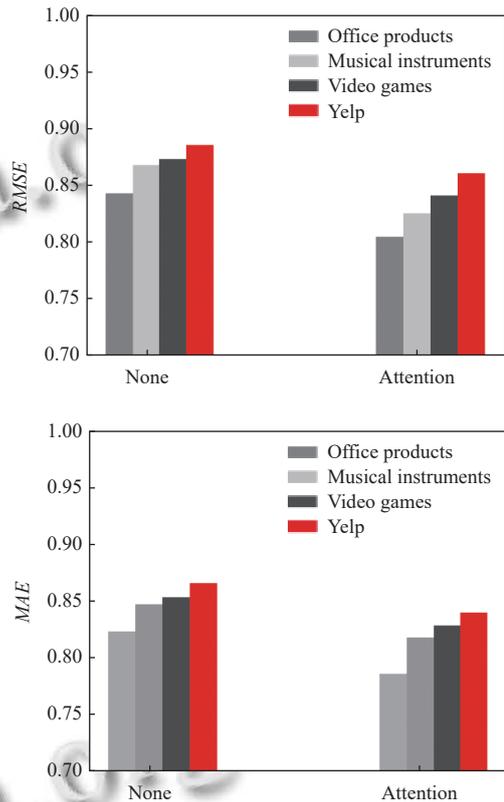


图6 Attention机制对模型性能的影响

由图6可知,MSFF模型在引入Attention机制后,评分预测误差明显降低了,说明注意力机制能够帮助SA_BiLSTM从评论文本中捕捉重要的信息,从而减少噪声和不相关信息的干扰.

4 结论与展望

本文提出的一种基于多源特征融合的推荐算法模型,通过评论文本作为辅助信息融入评分信息中提升评分预测的准确度.模型利用CNN和SA_BiLSTM分别提取评论文本中的用户特征和商品特征并将其融入隐语义模型得到的用户偏好和商品隐特征信息中得到更加完善的用户偏好和商品特征,然后将其输入全连接层中,从而预测得到用户可能对项目的评分,形成推

荐列表,从而达到推荐目的,同时也较好地解决了评分矩阵所面临的稀疏性问题.实验结果表明,本文所提出的模型,在用户-商品评分预测准确度较高,推荐效果较好.

参考文献

- 1 Liu XM, Wu SC, Zhang ZH, *et al.* Unify local and global information for top-N recommendation. Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. Madrid: ACM, 2022. 1262–1272.
- 2 郝志峰, 廖祥财, 温雯, 等. 基于多上下文信息的协同过滤推荐算法. 计算机科学, 2021, 48(3): 168–173. [doi: [10.11896/jsjx.200700101](https://doi.org/10.11896/jsjx.200700101)]
- 3 李昆仑, 林娜, 王珺. 融合评分与评论的深度评分预测模型. 小型微型计算机系统, 2023, 44(4): 737–744.
- 4 杨斌, 梁婧, 周佳薇, 等. 基于注意力机制的可解释点击率预估模型研究. 计算机科学, 2023, 50(5): 12–20. [doi: [10.11896/jsjx.221000032](https://doi.org/10.11896/jsjx.221000032)]
- 5 Gao C, Zheng Y, Li N, *et al.* A survey of graph neural networks for recommender systems: Challenges, methods, and directions. ACM Transactions on Recommender Systems, 2023, 1(1): 3.
- 6 Huang C, Chen JH, Xia LH, *et al.* Graph-enhanced multi-task learning of multi-level transition dynamics for session-based recommendation. Proceedings of the 35th AAAI Conference on Artificial Intelligence. AAAI, 2021. 4123–4130.
- 7 Zhao S, Du ZW, Chen J, *et al.* Hierarchical representation learning for attributed networks. IEEE Transactions on Knowledge and Data Engineering, 2021, 35(3): 2641–2656.
- 8 陈丽琼, 范国庆, 毕晓钰, 等. 融合评论文本与评分交互特征的推荐方法. 计算机工程与设计, 2023, 44(2): 393–399.
- 9 Zhang Q, Wang JW, Huang HR, *et al.* Hashtag recommendation for multimodal microblog using co-attention network. Proceedings of the 26th International Joint Conference on Artificial Intelligence. Melbourne: AAAI, 2017. 3420–3426.
- 10 Chen C, Zhang M, Liu YQ, *et al.* Neural attentional rating regression with review-level explanations. Proceedings of the 2018 World Wide Web Conference. Lyon: International World Wide Web Conferences Steering Committee, 2018. 1583–1592.
- 11 Lu YC, Dong RH, Smyth B. Coevolutionary recommendation model: Mutual learning between ratings and reviews. Proceedings of the 2018 World Wide Web Conference. Lyon: International World Wide Web Conferences Steering Committee, 2018. 773–782.
- 12 Chen X, Zhang YF, Qin Z. Dynamic explainable recommendation based on neural attentive models. Proceedings of the 33rd AAAI Conference on Artificial Intelligence. Honolulu: AAAI, 2019. 53–60.
- 13 Seo S, Huang J, Yang H, *et al.* Interpretable convolutional neural networks with dual local and global attention for review rating prediction. Proceedings of the 11th ACM Conference on Recommender Systems. Como: ACM, 2017. 297–305.
- 14 Da’u A, Salim N, Idris R. An adaptive deep learning method for item recommendation system. Knowledge-based Systems, 2021, 213: 106681. [doi: [10.1016/j.knosys.2020.106681](https://doi.org/10.1016/j.knosys.2020.106681)]
- 15 Chen XC, Li L, Pan WK, *et al.* A survey on heterogeneous one-class collaborative filtering. ACM Transactions on Information Systems (TOIS), 2020, 38(4): 35.
- 16 朵琳, 杨丙. 一种基于用户兴趣概念格的推荐评分预测方法. 小型微型计算机系统, 2020, 41(10): 2104–2108.
- 17 Zhang FZ, Yuan NJ, Lian DF, *et al.* Collaborative knowledge base embedding for recommender systems. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco: ACM, 2016. 353–362.
- 18 Paudel B, Bernstein A. Random walks with erasure: Diversifying personalized recommendations on social and information networks. Proceedings of the 2021 Web Conference. Ljubljana: ACM, 2021. 2046–2057.
- 19 Ma DH, Wang YF, Ma JH, *et al.* SGNR: A social graph neural network based interactive recommendation scheme for E-commerce. Tsinghua Science and Technology, 2023, 28(4): 786–798. [doi: [10.26599/TST.2022.9010050](https://doi.org/10.26599/TST.2022.9010050)]
- 20 Kim D, Park C, Oh J, *et al.* Convolutional matrix factorization for document context-aware recommendation. Proceedings of the 10th ACM Conference on Recommender Systems. Boston: ACM, 2016. 233–240.
- 21 Serra G, Tiño P, Xu Z, *et al.* An interpretable alternative to neural representation learning for rating prediction-transparent latent class modeling of user reviews. Proceedings of the 2024 International Joint Conference on Neural Networks (IJCNN). Yokohama: IEEE, 2024. 1–8.
- 22 Suryana N, Basari ASH. Generate contextual insight of product review using deep LSTM and word embedding. Journal of Physics: Conference Series, 2020, 1577(1): 012006. [doi: [10.1088/1742-6596/1577/1/012006](https://doi.org/10.1088/1742-6596/1577/1/012006)]

- 23 Shuai J, Zhang K, Wu L, *et al.* A review-aware graph contrastive learning framework for recommendation. Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval. Madrid: ACM, 2022. 1283–1293.
- 24 Liu LM, Cai L, Zhang C, *et al.* LinRec: Linear attention mechanism for long-term sequential recommender systems. Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. Taipei: ACM, 2023. 289–299.
- 25 Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems. *Computer*, 2009, 42(8): 30–37.
- 26 Zheng L, Noroozi V, Yu PS. Joint deep modeling of users and items using reviews for recommendation. Proceedings of the 10th ACM International Conference on Web Search and Data Mining. Cambridge: ACM, 2017. 425–434.
- 27 Xi WD, Huang L, Wang CD, *et al.* Deep rating and review neural network for item recommendation. *IEEE Transactions on Neural Networks and Learning Systems*, 2022, 33(11): 6726–6736. [doi: [10.1109/TNNLS.2021.3083264](https://doi.org/10.1109/TNNLS.2021.3083264)]
- 28 Chen C, Zhang M, Liu Y, *et al.* Neural attentional rating regression with review-level explanations. Proceedings of the 2018 World Wide Web Conference. 2018. 1583–1592. [doi: [10.1145/3178876.3186070](https://doi.org/10.1145/3178876.3186070)]
- 29 潘莹, 李浩, 王世辉, 等. 融合注意力机制和评论文本特征的推荐算法. *计算机工程与设计*, 2024, 45(9): 2764–2770.
- 30 钱忠胜, 赵畅, 俞情媛, 等. 结合注意力 CNN 与 GNN 的信息融合推荐方法. *软件学报*, 2023, 34(5): 2317–2336. [doi: [10.13328/j.cnki.jos.006405](https://doi.org/10.13328/j.cnki.jos.006405)]

(校对责编: 张重毅)