

融合评论文本和评分矩阵的电影推荐算法研究

张露怡¹, 余敦辉^{1,2}

¹(湖北大学 计算机与信息工程学院, 武汉 430062)

²(湖北省教育信息化工程技术中心, 武汉 430062)

E-mail: 201822111920042@stu.hubu.edu.cn

摘要: 针对电影推荐领域评论文本信息在传统推荐中未被充分利用, 推荐准确度不高的问题, 提出一种融合评论文本和评分矩阵的电影推荐算法. 首先通过基于自注意力机制的双向门控循环单元神经网络对电影的高质量评论文本进行建模, 提取评论文本中的特征, 构建电影评论特征矩阵. 同时使用隐语义模型对用户评分矩阵进行分解, 得到用户潜在兴趣矩阵和电影潜在特征矩阵. 最后通过改进的 DeepFM 融合电影评论特征矩阵和电影潜在特征矩阵得到电影综合得分并形成推荐列表, 以达到推荐的目的. 实验结果表明, 与其他几种电影推荐算法相比, 在 AUC、F-Score、RMSE 上平均提升分别约 7.37%、9.32%、8.23%, 最高提升分别为 11.60%、15.22%、12.79%.

关键词: 推荐算法; 门控循环单元; 自注意力机制; 评分矩阵; 隐语义模型

中图分类号: TP391

文献标识码: A

文章编号: 1000-1220(2022)10-2063-07

Research on Movie Recommendation Algorithm Combining Review and Rating Matrix

ZHANG Lu-yi¹, YU Dun-hui^{1,2}

¹(College of Computer and Information Engineering, Hubei University, Wuhan 430062, China)

²(Education Informationization Engineering and Technology Center, Wuhan 430062, China)

Abstract: In the field of movie recommendation, focusing on the problems that the movie review information was underutilized in traditional movie recommendation which not led to high recommendation accuracy, a movie recommendation algorithm combining movie review and rating matrix was proposed. Firstly, the bidirectional gated recurrent unit neural network based on self-attention mechanism was used to model the high-quality movie reviews, extract the features from them, and construct the feature matrix of movie review. At the same time, the latent factor model was used to decompose the rating matrix to get the potential interest matrix of users and the potential feature matrix of movie. At last, improved DeepFM was used to combine the feature matrix of movie review and the potential feature matrix of movie. Through the above process, the movie comprehensive score was calculated and the movie recommendation list was formed. The experimental results show that compared with some other recommendation algorithms, the average increase in AUC, F-score and RMSE is about 7.37%, 9.32% and 8.23%, and the highest increase is 11.60%, 15.22% and 12.79%.

Key words: recommendation algorithm; gated recurrent unit; self-attention mechanism; scoring matrix; latent factor model

1 引言

随着互联网的蓬勃发展, 面对爆炸式增长的数据规模, 用户如果仅依靠简单的手动搜索, 将难以挑选到自己感兴趣的物品. 为提升用户的消费体验, 推荐系统^[1,2]应运而生. 由于推荐系统能够精准定位用户兴趣, 实现信息产生者和信息消费者的完美匹配, 所以在网络服务中的实际应用也越来越多, 如网易云音乐根据用户播放和收藏为用户生成“每日推荐”列表; 豆瓣根据用户打分和观看过的电影为其推荐可能感兴趣的电影等等.

在电影评论领域, 有众多高质量评论网站, 如 IMDB, ROTTEN TOMATOES 等. 但是大部分电影推荐系统研究仅仅分析用户和电影的历史交互信息, 没有充分发挥这些网站中优秀评论文本的内在价值. 因此本文将利用电影的高质量

评论文本, 通过基于自注意力机制的双向门控循环单元神经网络提取评论文本中的特征, 构建电影评论特征矩阵. 同时, 为在电影评分数据中获得用户的兴趣偏好, 使用隐语义模型对用户评分矩阵进行分解, 得到用户潜在兴趣矩阵和电影潜在特征矩阵. 最后通过改进的 DeepFM 融合电影评论特征矩阵和电影潜在特征矩阵得到最终电影综合得分并形成推荐列表, 以达到推荐的目的.

2 相关工作

当前, 电影推荐受到了众多学者和研究人员的关注. 协同过滤算法首先被应用于电影推荐之中. 文献[3]将深度潜在因子模型(DLFM, Deep Latent Factor Model)用于基于协作过滤的算法推荐系统中. 文献[4]根据电影推荐中的各种隐式

反馈,并将这些反馈纳入协作过滤.但是,基于协同过滤的算法具有冷启动问题,且对用户的评分数据依赖性较大.为解决以上问题,基于内容的推荐算法逐渐被采用.文献[5]采用基于内容的推荐方法,使用目标编程模型来预测缺少的电影评分.文献[6]提出一种新颖的基于内容的推荐方法,即在向用户推荐商品时使用多属性网络来计算相关性,从而有效地反映多个属性.文献[7]则通过考虑用户偏好来提取隐藏在内容特征中的关系的优化模型,然后将学习到的特征关系矩阵应用于推荐,并解决冷启动问题.此类算法在一定程度上缓解了冷启动问题,但受限于新用户-新产品的限制,且具有新颖性不强的问题.

随着对深度学习的深入研究,该技术也广泛运用于电影推荐,并取得成功.文献[8]提出卷积矩阵分解模型,解决用户对项目评分数据的稀疏性问题,并额外利用了文本数据提高了推荐的准确性.文献[9]利用深度学习来学习用户和电影特征信息,以形成高维潜在空间的用户特征和电影特征的隐向量.文献[10]则通过神经网络对“灰羊用户”,即具有独特偏好和品味的用户推荐电影.基于深度学习方法进行电影推荐大大提升了推荐的精准度和效率,但其推荐结果缺乏合理的解释性的问题也深受诟病.

3 融合评论文本与评分矩阵的电影推荐算法

本文提出了一种融合评论文本与评分矩阵的电影推荐算法(CRRM,Combining Review and Rating Matrix),算法整体结构如图1所示,具体流程如下:

- 1)提出基于自注意力机制的双向门控循环单元神经网络(SA-BiGRU,Self-Attention-Bidirectional Gated Recurrent Unit)模型提取电影评论中的特征,构建电影评论特征矩阵.
- 2)通过隐语义模型(LFM,Latent Factor Model)分解电影评分矩阵,得到用户潜在兴趣矩阵和电影潜在特征矩阵.
- 3)通过改进的DeepFM融合电影评论特征矩阵与电影潜在特征矩阵并计算最终的电影得分.

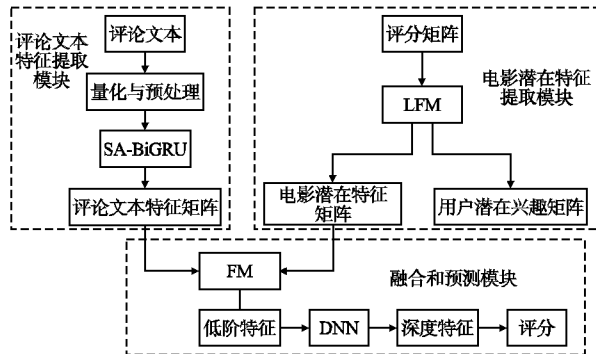


图1 CRRM算法结构图

Fig.1 CRRM algorithm structure diagram

此算法使SA-BiGRU模型和LFM充分发挥了各自的优势,克服了单一模型的缺陷,提高了推荐结果的可解释性和多样性.同时充分利用电影评论文本和电影评分数据,从而提升最终的推荐结果.

算法中所用重要符号及其说明如表1所示.

3.1 基于SA-BiGRU的电影评论特征提取

很多电影拥有大量的评分数据,但评分数据仅能反映用户对电影的整体满意度,而缺乏电影具体特征和用户偏好等信息.所以需要引入高质量评论文本,更好地获取用户对电影的个性化偏好,实现面对用户的精准化个性推荐.

表1 符号对照说明表

Table 1 Symbol contrast explanatory table

符号	符号说明
z_t, r_t	更新门控信号,重置门控信号
$\tilde{h}_t, \vec{h}_t, \overleftarrow{h}_t, h_t$	当前时刻的候选激活状态,单向激活状态,反向激活状态,激活状态
Q, K, V	查询向量序列,键向量序列,值向量序列
W_Q, W_K, W_V	查询向量序列参数矩阵,键向量序列参数矩阵,值向量序列参数矩阵
d_q, d_k, d_v	矩阵 W_Q 的维度,矩阵 W_K 的维度,矩阵 W_V 的维度
H	电影评论总特征矩阵
U, V, R	用户潜在兴趣矩阵,电影潜在特征矩阵,评分矩阵
$H_{feature}^1, H_{feature}^2$	电影评论低阶特征,电影评论深度特征
$V_{feature}^1, V_{feature}^2$	电影潜在低阶特征,电影属性深度特征
\hat{R}_{HV}	电影最终评分

文献[11]使用长短期记忆神经网络LSTM对评论文本数据进行建模,得到了较为满意的结果.而门控循环单元GRU作为LSTM的一个变体,在保持LSTM效果的同时,计算简便,结构简单,易于收敛,且因为在序列处理上具有出色表现而被广泛应用于自然语言处理任务中.因此本文结合双向门控循环单元与自注意力机制提出SA-BiGRU模型通过电影评论文本提取特征^[12].

SA-BiGRU模型一共包括5层网络结构,如图2所示.

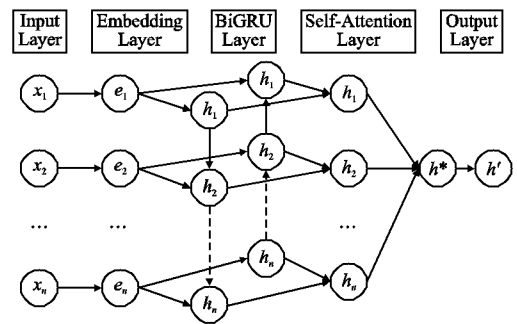


图2 SA-BiGRU网络结构图

Fig.2 SA-BiGRU network structure diagram

- 1)输入层:将量化后的电影评论文本输入到模型中.
- 2)嵌入层:产生每个单词在低维空间上的映射.
- 3)BiGRU层:提取词汇特征.
- 4)Self-Attention层:将词汇特征与权重向量相乘的结果进行迭代,产生句子特征.
- 5)输出层:输出求得电影评论特征矩阵.

3.1.1 电影评论文本量化

为将电影评论文本转换为神经网络可以处理的数据结构,首先将评论文本转换为单词序列 $X = \{x_1, x_2, \dots, x_T\}$,并对每个单词 x_i 进行独热编码,即将每个单词转化为实数向

量. 向量的每个元素代表词库中的一个单词, 只有单词 x_i 对应的元素值为 1, 其余均为 0. 为满足神经网络输入必须行列相同的要求, 根据所有文本的情况设定一个合适的统一列数, 列数不够的矩阵用 0 向量填充. 通过以上转换, 每个评论都可以转换成一个等行列数的单词序列矩阵 X .

3.1.2 评论文本输入与预处理

将评论文本量化后得到的单词序列 X 作为 SA-BiGRU 模型的输入. 这里的单词序列 X 将严格按照评论文本的单词顺序以保持时序关系, 为后面的处理做了准备.

由于用户对电影的评论原始文本的数据集通常比较大, 其单词表的数量和单词 x_i 的维度可能有几百万维甚至更多. 因此, 每个单词 x_i 需经过 SA-BiGRU 模型的嵌入层降维为低维实数向量 e_i . 转化过程如下:

$$e_i = W^{word} x_i \quad (1)$$

其中词向量矩阵为 $W^{word} \in \mathbb{R}^{d^{word} \times |V|}$, d^{word} 是评论文本本身词向量的维度, $|V|$ 是单词 x_i 的维度, W^{word} 则是网络中学习到的参数. 转换后的向量维度得到了大幅度的缩减, 大概为几百维, 减少了神经网络中的参数规模, 提高了网络的学习效率.

3.1.3 电影评论特征矩阵生成

本文通过 BiGRU 层对经过量化和降维之后的评论文本特征进行提取, 其中包含了更新门和重置门两个门限.

更新门是决定上一时刻的状态信息传递到当前时刻的程度. 求得更新门控信号 z_t 为:

$$z_t = \sigma(W^{(z)} x_t + U^{(z)} h_{t-1}) \quad (2)$$

重置门则是控制上一时刻的状态信息被遗忘的程度. 求得重置门控信号 r_t 为:

$$r_t = \sigma(W^{(r)} x_t + U^{(r)} h_{t-1}) \quad (3)$$

其中 x_t 表示当前时刻输入的单词向量, h_{t-1} 表示前一个时刻的单词信息, W 表示输入的单词权重, U 表示循环输入的单词权重.

得到门控信号 z_t 和 r_t 之后, 再通过重置门来得到遗忘后的状态 $h_{t-1} \otimes r_t$, 其中 \otimes 表示哈达马积对应元素相乘. 然后, 将遗忘后的状态 $h_{t-1} \otimes r_t$ 与当前时刻的单词向量输入 x_t 相加. 最后通过非线性函数 \tanh 激活得到当前时刻的候选激活状态为:

$$\tilde{h}_t = \tanh(W_h x_t + U_h (h_{t-1} \otimes r_t)) \quad (4)$$

最后, 通过更新门对当前节点的输入选择记忆. 求得当前时刻的激活状态为:

$$\vec{h}_t = (1 - z_t) \otimes h_{t-1} + z_t \otimes \tilde{h}_t \quad (5)$$

由于评论文本具有时序关系, 因此当前时刻的前向信息和后向信息都非常重要. 本文采用了 BiGRU 模型以增强网络对时序关系的获取, 在原有的正向 GRU 网络层的基础上增加一层反向的 GRU 层, 以解决单向 GRU 不能捕获后向信息的问题. 此时当前时刻的激活状态可以表示成:

$$h_t = [\vec{h}_t, \overleftarrow{h}_t] \quad (6)$$

为解决在电影评论文本的单词序列中不同单词对整个句子产生不同的影响的问题, 本文引入了注意力机制. 而自注意力机制 Self-Attention^[13] 作为注意力机制的一个变体, 更容易捕获句子中远距离的相互依赖的特征, 在计算过程中将句子

中任意两个单词通过一个计算步骤直接联系起来, 所以远距离依赖特征的距离被极大缩短, 更加有效地利用这些特征. 因此在 BiGRU 层后添加一层 Self-Attention 层, 对权重高的单词给予更多的关注, 而不重要的单词则消耗更少的资源, 以提高网络的效率, 同时得到更有价值的特征表示.

将 BiGRU 层输出的向量集合 h_t 表示为 $H_t = \{h_{t_0}, h_{t_1}, \dots, h_{t_N}\}$. Self-Attention 层通过线性变换得到 3 组向量序列 Q, K, V :

$$\begin{aligned} Q &= W_Q H_t \in \mathbb{R}^{d_q \times N} \\ K &= W_K H_t \in \mathbb{R}^{d_k \times N} \\ V &= W_V H_t \in \mathbb{R}^{d_v \times N} \end{aligned} \quad (7)$$

其中 Q, K, V 分别为查询向量序列, 键向量序列和值向量序列, W_Q, W_K, W_V 分别为学习到的查询向量序列, 键向量序列和值向量序列的参数矩阵, d_q, d_k, d_v 分别为矩阵 W_Q, W_K, W_V 的维度.

最终提取的语句特征矩阵 h^* 如下所示:

$$h^* = \sum_{j=1}^N a_j v_j \quad (8)$$

$$a = \text{soft max}(s(k_j, q_i)) v_j \quad (9)$$

其中 $i, j \in [1, N]$ 为输出和输入向量序列的位置, 连接权重 a_{ij} 由注意力机制动态生成.

最后通过一个全连接的输出层. 基于自注意力机制的门控循环单元网络输入用户对电影的评论文本, 提取到特征维度为 k 的电影评论特征矩阵为:

$$h' = \text{SA-BiGRU}(\omega, \text{Review}) \quad (10)$$

$h' \in \mathbb{R}^{k \times n}$ 的列表表示评论文本, 行表示评论特征, ω 为该网络中所有权重和偏置参数, Review 为原始评论文本.

对于同一电影而言会存在多条高质量评论文本, 每一条评论文本经过以上过程后会生成不同的潜在特征 h' . 为全面得出从评论文本中提取到的电影总特征矩阵, 对 n 个特征 h' 求均值得到电影的评论总特征矩阵为:

$$H = \frac{1}{n} \sum_{i=1}^n h'_i \quad (11)$$

3.2 基于 LFM 的电影潜在特征矩阵生成

隐语义模型 (LFM) 作为一种有效的特征映射方法, 可以精确描述用户兴趣偏好和物品本身特性, 因此在推荐系统中得到了广泛应用. 本文通过 LFM 模型分解电影的评分矩阵, 得到用户潜在兴趣矩阵和电影潜在特征矩阵. 具体流程如图 3 所示.

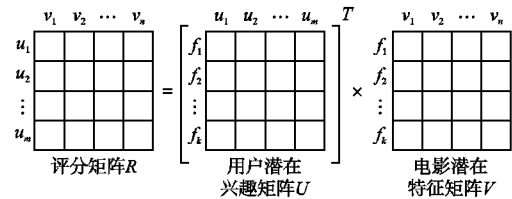


图 3 LFM 框架

Fig. 3 Latent factor model framework

假设有 m 个用户, 用户集合为 $U = \{u_1, u_2, \dots, u_m\}$, 对 n 部电影, 电影集合为 $V = \{v_1, v_2, \dots, v_n\}$ 进行评分, 评分矩阵 R 代表用户对电影的评分矩阵. 而 LFM 的目标是通过评分矩阵

R 找到两个矩阵 U 和 V , 使两个矩阵的乘积近似等于评分矩阵 R .

$$V \approx UR \quad (12)$$

其中 U 和 V 分别为用户潜在兴趣矩阵和电影潜在特征矩阵.

为获得更为精确的电影潜在特征矩阵, 接下来通过最小化损失函数迭代获得用户潜在兴趣矩阵和电影潜在特征矩阵. 损失函数的表达式如下所示:

$$L(U, V) = \sum_u \sum_r (R_{ur} - U_u^T V_r)^2 + \lambda (\sum_u \|U_u\|^2 + \sum_r \|V_r\|^2) \quad (13)$$

公式的前半部分表示所有评分预测值与实际值的偏差,

其中 R_{ur} 为真实值, $U_u^T V_r$ 为预测值, $\lambda (\sum_u \|U_u\|^2 + \sum_r \|V_r\|^2)$ 是为了避免训练过拟合而引入的正则化项, λ 为正则化参数. 本文通过随机梯度下降法求解上述损失函数: 首先对参数求偏导, 找到梯度下降最快的方向, 然后通过迭代不断地优化参数直至收敛即可求得电影潜在特征矩阵 V .

3.3 电影潜在特征与评论特征融合

本文改进 DeepFM^[14] 中设计的融合层来融合电影评论特征矩阵与电影潜在特征矩阵, 并求得电影综合评分, 形成最终的推荐列表. 具体步骤如下:

1) 将电影评论特征矩阵 H 输入因子分解模型 (FM, Factorization Machine) 中计算出电影评论低阶特征 $H_{feature}^1$:

$$H_{feature}^1 = \hat{w}_0 + \sum_{i=1}^{1M} \hat{w}_i H_i + \frac{1}{2} \left(\left(\sum_{i=1}^{1M} H_i \right)^2 - \sum_{i=1}^{1M} (H_i)^2 \right) \quad (14)$$

其中 \hat{w}_0 表示全局偏差, \hat{w}_i 表示 H 中第 i 个变量的权重.

2) 同理基于电影潜在特征矩阵 V 计算得到电影潜在低阶特征 $V_{feature}^1$.

3) 将电影评论低阶特征 $H_{feature}^1$ 输入到前向的深度神经网络 (DNN, Deep Neural Network) 里计算出电影评论深度特征 $H_{feature}^2$:

$$\begin{aligned} H_{feature}^{2(1)} &= w^{(1)} H^{(1)} + b^{(1)} \\ H_{feature}^{2(2)} &= \sigma^{(2)} (w^{(2)} H_{feature}^{2(1)} + b^{(2)}) \\ &\vdots \\ H_{feature}^{2(L)} &= \sigma^{(L)} (w^{(L)} H_{feature}^{2(L-1)} + b^{(L)}) \end{aligned} \quad (15)$$

其中, $H_{feature}^{2(i)}$ 为第 i 层的输出向量, $\sigma^{(i)}$ 为第 i 层的激活函数, 这里选用 Relu 激活函数, $w^{(i)}$ 为第 i 层的权重向量, $b^{(i)}$ 为第 i 层的偏差向量, L 代表隐藏层的层数.

4) 同理基于电影潜在低阶特征 $V_{feature}^1$ 计算得到电影属性深度特征 $V_{feature}^2$.

5) 使用一个全连接层将 $H_{feature}^2$ 和 $V_{feature}^2$ 进行组合, 得到电影最终评分 \hat{R}_{HV} :

$$\hat{R}_{HV} = w \times (H_{feature}^2 \otimes V_{feature}^2) + b_h + b_s + \mu \quad (16)$$

其中 w 表示权重矩阵, b_h 表示电影评论偏差, b_s 表示电影评分偏差, μ 表示全局偏差, \otimes 表示特征向量相对应的元素相乘.

6) 根据得到的电影综合评分 \hat{R}_{HV} 形成最终的推荐列表.

3.4 算法执行过程

算法. CRRM (Combining Review and Rating Matrix) 算法

输入: 电影评论文本的单词序列 $X = \{x_1, x_2, \dots, x_T\}$

输出: 电影评分 \hat{R}_{HV}

1. for each 单词序列 X :

2. for each 单词 x_i :

3. 降维 $e_i = W^{word} x_i$

4. BiGRU 层得评论词汇特征 h_i

5. 评论特征 H 集合通过自注意力层得到句子特征 $h^* = \sum_{j=1}^N a_{ij} v_j$

6. 全连接输出层得到电影评论特征矩阵 $h' = SA - BiGRU(W, Review)$

7. 求均值得到电影总评论特征矩阵 $H = \frac{1}{n} \sum_{i=1}^n h'_i$

8. LFM 模型分解电影评分矩阵得到电影潜在特征矩阵 $V \approx UR$

9. 改进 DeepFM 融合 H 和 V 得电影综合分 $\hat{R}_{HV} = w \times (H_{feature}^2 \otimes V_{feature}^2) + b_h + b_s + \mu$

10. return 电影推荐列表.

4 实验结果分析

4.1 数据集

本文数据集来自 MovieLens. 该数据集是以研究为目的的数据集, 包含多用户对多部电影的评级数据, 也包括电影元数据信息和用户属性信息. 同时, 本文以数据集中包含的电影链接为入口, 在 IMDB 上爬取电影链接的所有评论. 实验前还需要对每个评论文本数据采用以下预处理的方式进行处理:

1) 去除一些无意义的评论, 如只包含符号的无用评论等;

2) 对评论文本进行大小写转化, 将所有大写字母转化为小写;

3) 根据空格进行分词;

4) 去停用词, 删除没有意义的符号和单词;

5) 将所有单词处理成原形.

由于部分电影拥有大量的评论, 为节省资源, 提高模型的效率, 设置评论数量最大值 θ , 超出的部分将被删除. 为同时满足保证评论文本的质量和输入序列长度相同两个条件, 本文只选择评论词数在 50-300 之间的评论文本, 一共统计了 5532 部电影, 共计 337452 条评论, 5675832 个评分. 将数据集随机以 8:1:1 切分为训练集、交叉验证集和测试集进行实验.

4.2 评价指标

4.2.1 AUC

AUC 衡量了模型对样本正确排名的能力, 其值越大, 说明模型的排名能力越高.

$$AUC = \frac{\sum_{ins_j \in positiveclass} rank_{ins_j} - \frac{M \times (M + 1)}{2}}{M \times N} \quad (17)$$

其中 $rank_{ins_j}$ 指第 i 条样本的序号, M 和 N 分别指正样本个数和负样本个数, $\sum_{ins_j \in positiveclass}$ 指正样本的数量相加.

4.2.2 F-Score

F 值将准确率 (Precision) 和召回率 (Recall) 加权平均, 均匀地反映了推荐效果, 其中准确率反映推荐算法的推荐水平, 召回率反映被推荐算法中物品占用户真正喜欢物品的比重:

$$Precision = \frac{TP}{TP + FP} \quad (18)$$

$$Recall = \frac{TP}{TP + FN} \quad (19)$$

$$F - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (20)$$

其中 TP 指用户喜欢且推荐的电影, FN 指用户不喜欢但推荐的电影, FP 指用户喜欢但没推荐的电影.

4.2.3 RMSE

RMSE 用于评价模型的预测精确度. 其值越小, 说明预测模型具有更好的精确度.

$$RMSE = \sqrt{\frac{1}{|T|} \sum_{(u,i) \in T} (\hat{r}_{ui} - r_{ui})^2} \quad (21)$$

其中 T 指测试集, \hat{r}_{ui} 指模型预测出来的评分, r_{ui} 指测试集的实际评分.

4.3 对比模型

为评估本文提出的 CRRM 模型有效性, 将其与以下 5 种方法进行对比:

1) RNN^[15]: 递归神经网络 (Recursive Neural Network) 通过建模用户的历史序列行为来预测用户的下一个动作.

2) GAN^[16]: 生成式对抗网络 (Generative Adversarial Networks) 包括基于生成式的网络学习模型和基于判别式的网络学习模型两个模块, 通过两个模型的相互博弈来学习用户的历史数据并排列出前 k 个用户感兴趣的项.

3) DeepCoNN^[17]: 深度协同神经网络 (Deep Cooperative Neural Networks) 通过两个并行的神经网络将用户评论集和项目评论集结合起来对用户和项目进行建模.

4) SA-GRU; SA-BiLSTM 模型与 SA-BiGRU 模型属于同种类型的混合实验模型. SA-BiLSTM 模型除了将 SA-BiGRU 模型中的 BiGRU 层换成 GRU 层, 其他实验设置均相同.

5) SA-BiLSTM; SA-BiLSTM 模型与 SA-BiGRU 模型属于同种类型的混合实验模型. SA-BiLSTM 模型除了将 SA-BiGRU 模型中的 BiGRU 层换成 BiLSTM 层, 其他实验设置均相同.

4.4 实验结果分析

4.4.1 实验环境

实验环境如表 2 所示.

表 2 实验环境

Table 2 Experimental environment

实验环境	实验配置
Operating systems	Ubuntu 16.04
CPU	Intel (R) Core (TM) i7-10510U CPU @ 1.80GHz 2.30GHz
RAM	16G
Programming Language	Python3.7
Deep Learning Framework	Tensorflow + Keras

4.4.2 网络超参数

网络超参数如表 3 所示.

表 3 网络超参数

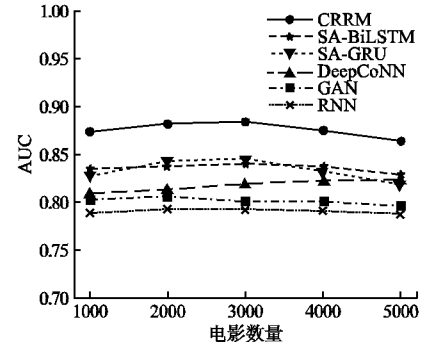
Table 3 Experimental environment

网络参数	参数值
learning_rate	0.2
output_keep_prob	0.8
num_layers	5
max_gradient_norm	5
dropout	0.5

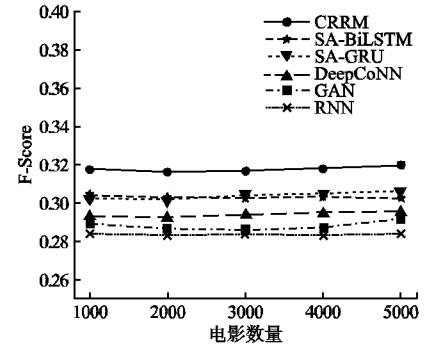
4.4.3 算法有效性验证

为同时融合评论文本和评分数据对算法有效性进行验证, 本文从电影数量、评论文本数量和评分数量 3 个维度, 将所提出的算法与其他该领域主流算法进行对比实验.

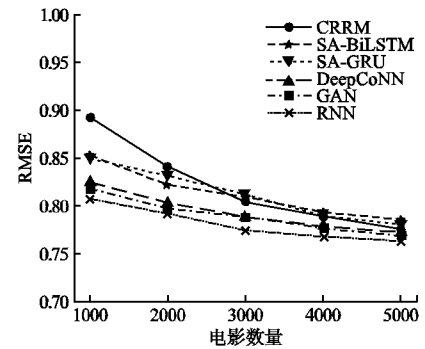
1) 电影数量对算法性能的影响 (单个电影的高质量评论文本数量设定为 50, 评分数量设定为 1000).



(a) 不同电影数量下的 AUC



(b) 不同电影数量下的 F-Score



(c) 不同电影数量下的 RMSE

图 4 不同电影数量下的实验结果

Fig. 4 Experimental result under different number of movies

从图 4 可以看出, 算法随着电影数量的增多, 在 AUC 和 F 值上变化较为平稳, 且较其他 5 种算法的最大提升比率在 8% 左右. 在电影数量小于 2000 时 RMSE 的相较于其他算法提升较为明显, 因为 SA-BiGRU 中的 BiGRU 层相较于 LSTM 以及其他方法, 参数少, 收敛更容易, 因而在数据量规模不大时, 优势更为明显, 提升比率在 7% 左右, 具体如表 4 所示.

2) 高质量评论文本数量对算法性能的影响 (电影数量设定为 5000, 评分数量设定为 1000).

从图 5 可以看出, 算法随着评论数量的增多, 在 AUC 和

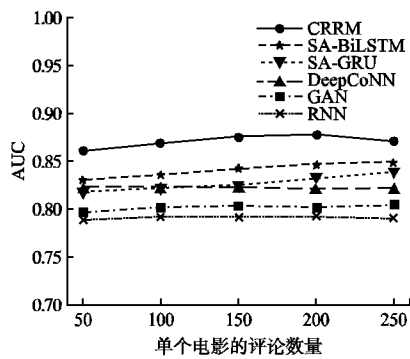
F值上有微小的上升趋势,且较其他5种算法的提升比率在7%和10%左右, RMSE呈明显上升趋势,在单个电影评论数

表4 相对其他算法的性能提升比率(1)

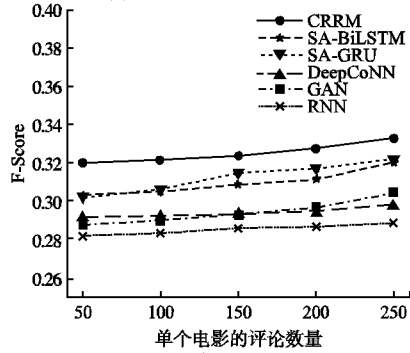
Table 4 Performance improvement ratio 1 compared with other algorithms

Algorithm	Maximum improvement ratio of AUC	Maximum improvement ratio of F-Score	Maximum improvement ratio of RMSE
A-BiLSTM	5.31	5.66	4.66
A-GRU	5.62	5.10	5.02
DeepCoNN	8.49	8.43	8.30
DeepFM	10.48	10.77	9.22
LFM	11.60	12.47	10.64

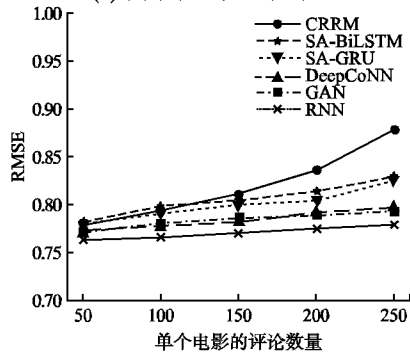
量大于40时相对其他算法提升较为明显,提升比率在9%左右,具体如表5所示。



(a)不同评论数量下的AUC



(b)不同评论数量下的F-Score



(c)不同评论数量下的RMSE

图5 不同评论数量下的实验结果

Fig. 5 Experimental result under different number of reviews

3)评分数量对算法性能的影响(电影数量设定为5000,

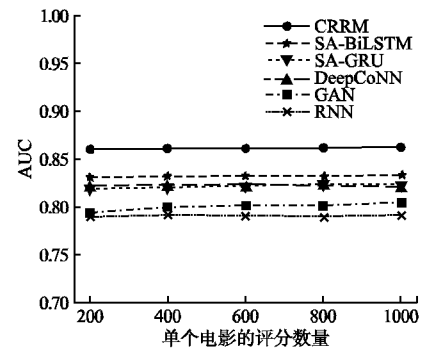
单个电影的高质量评论文本数量设定为50)。

表5 相对其他算法的性能提升比率(2)

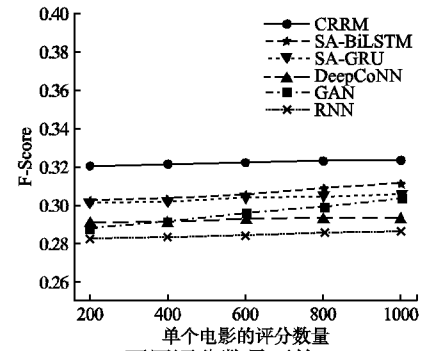
Table 5 Performance improvement ratio 2 compared with other algorithms

Algorithm	Maximum improvement ratio of AUC	Maximum improvement ratio of F-Score	Maximum improvement ratio of RMSE
A-BiLSTM	4.03	5.54	5.92
A-GRU	6.25	6.03	6.55
DeepCoNN	6.88	11.48	10.18
DeepFM	9.57	11.18	10.76
LFM	10.77	15.22	12.7

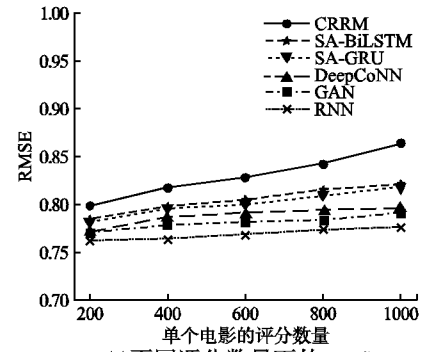
从图6可以看出,算法随着评分数量的增多,在AUC和F值上变化较为平稳,在RMSE上有小幅度的提升,因为本



(a)不同评分数量下的AUC



(b)不同评分数量下的F-Score



(c)不同评分数量下的RMSE

图6 不同评分数量下的实验结果

Fig. 6 Experimental result under different number of rates

算法融合了 LFM 分解评分矩阵后得到的潜在特征矩阵,算法性能更优,相对其他5种算法,提升比率分别在6%、9%和

8%左右,具体如表6所示。

表6 相对其他算法的性能提升比率(3)

Table 6 Performance improvement ratio 3 compared with other algorithms

Algorithm	Maximum improvement ratio of AUC	Maximum improvement ratio of F-Score	Maximum improvement ratio of RMSE
A-BiLSTM	3.68	5.90	5.12
A-GRU	5.16	6.67	5.53
DeepCoNN	5.26	10.34	8.42
DeepFM	8.35	11.18	9.07
LFM	9.09	13.88	11.24

5 结语

本文提出一种融合评论文本和评分矩阵的电影推荐算法,利用电影的高质量评论文本与评分矩阵使推荐结果更加准确全面。通过基于自注意力机制的双向门控循环单元神经网络对评论文本数据进行建模,得到电影评论特征矩阵,同时使用改进的 LFM 对用户评分矩阵进行分解,得出电影潜在特征矩阵。并通过改进的 DeepFM 将其与电影评论特征矩阵融合得到电影综合得分,形成推荐列表,以达到推荐的目的。实验结果表明,本文提出的电影推荐算法能够取得较好的推荐效果。由于电影评论更偏向于感性层面,本文未来将着重于对电影评论文本中的情感分析,以使推荐电影的准确性和完善性得到进一步提高。

References:

- [1] Huang Li-wei, Jiang Bi-tao, Lv Shou-ye, et al. Survey on deep learning based recommender systems[J]. Chinese Journal of Computers, 2018, 41(7): 1619-1647.
- [2] Zhou Wan-zhen, Cao Di, Xu Yun-feng, et al. A survey of recommendation systems[J]. Journal of Hebei University of Science and Technology, 2020, 41(1): 76-87.
- [3] Mongia A, Jhamb N, Chouzenoux E, et al. Deep latent factor model for collaborative filtering[J]. Signal Processing, 2019, 169: 107366.
- [4] Yutian Hu, Fei Xiong, Dongyuan Lu, et al. Movie collaborative filtering with multiplex implicit feedbacks[J]. Neurocomputing, 2020, 398: 485-494.
- [5] Emrah I, Fatih T, Cemalettin O. Moreopt: a goal programming based movie recommender system[J]. Journal of Computational Science, 2018, 28: S1877750317314540.
- [6] Son J, Kim S B. Content-based filtering for recommendation systems using multiattribute networks[J]. Expert Systems with Applications an International Journal, 2017, 89(dec. 15): 404-412.

- [7] Haijun, Zhang, Yanfang, et al. Bridging user interest to item content for recommender systems: an optimization model[J]. IEEE Transactions on Cybernetics, 2019, 50(10): 4268-4280.
- [8] Zheng L, Noroozi V, Yu P S. Joint deep modeling of users and items using reviews for recommendation[C]//10th ACM International Conference, ACM, 2017: 425-434.
- [9] Deng Cun-bin, Yu Hui-qun, Fan Gui-sheng. Integrating dynamic collaborative filtering and deep learning for recommendation[J]. Computer Science, 2019, 46(8): 28-34.
- [10] Alabdulrahman R, Viktor H. Catering for unique tastes: targeting grey-sheep users recommender systems through one-class machine learning[J]. Expert Systems with Applications, 2021, 166: 114061.
- [11] Fan Ru-xin. Research on personalized e-commerce product recommendation based on deep neural network[D]. Hefei: Hefei University of Technology, 2019.
- [12] Che L, Yang X, Wang L. Text feature extraction based on stacked variational autoencoder[J]. Microprocessors and Microsystems, 2020, 76: 103063, doi: 10.1016/j.micpro.2020.103063.
- [13] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in Neural Information Processing Systems, 2017, 30, arXiv: 1706.03762.
- [14] Wu Ying-ying. Hybrid recommendation algorithm based on rating matrix and review text[D]. Baoding: Hebei University, 2020.
- [15] Feng Yong, Zhang Bei, Qiang Bao-hua, et al. MN-HDRM: a novel Hybrid dynamic recommendation model based on long-short-term interests multiple neural networks[J]. Chinese Journal of Computers, 2019, 42(1): 16-28.
- [16] Chae D K, Kang J S, Kim S W, et al. CFGAN: a generic collaborative filtering framework based on generative adversarial networks[C]//27th ACM International Conference, ACM, 2018: 137-146.
- [17] Zheng L, Noroozi V, Yu P S. Joint deep modeling of users and items using reviews for recommendation[C]//10th ACM International Conference, ACM, 2017: 425-434.

附中文参考文献:

- [1] 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.
- [2] 周万珍, 曹迪, 许云峰, 等. 推荐系统研究综述[J]. 河北科技大学学报, 2020, 41(1): 76-87.
- [9] 邓存彬, 虞慧群, 范贵生. 融合动态协同过滤和深度学习的推荐算法[J]. 计算机科学, 2019, 46(8): 28-34.
- [11] 范汝鑫. 基于深度神经网络的个性化商品推荐研究[D]. 合肥: 合肥工业大学, 2019.
- [14] 吴莹莹. 基于评分矩阵和评论文本的混合推荐算法[D]. 保定: 河北大学, 2020.
- [15] 冯永, 张备, 强保华, 等. MN-HDRM: 长短兴趣多神经网络混合动态推荐模型[J]. 计算机学报, 2019, 42(1): 16-28.