

# 一种融合时间权值和用户行为序列的电影推荐模型

程 思, 陶宏才

(西南交通大学计算机与人工智能学院, 四川 成都 611756)

**摘要:**对用户历史观影行为进行研究可以给电影推荐提供意见和参考。在实际推荐应用中,随着时间的推移,用户的兴趣会发生相应的变化,而愈接近当前时刻的历史行为对当前时刻的兴趣状态的影响愈大。但已有的 GRU 模型只能通过历史行为推断出最终的兴趣,并未考虑到历史行为对最终兴趣的影响存在随时间衰减这一特性。文中提出 TGRU 改进模型将时间权值与 GRU 模型相融合,以此获得更加精确的兴趣表达。此外,使用 AUGRU 模型捕捉与最终兴趣相关的那些兴趣的演化过程,从而预测用户未来的兴趣。基于上述成果与分析,最终提出一种融合时间权值和用户行为序列的电影推荐模型——MRTUB,该模型较为全面地考虑了用户兴趣的提取及演化过程。实验结果表明,所提出的 MRTUB 电影推荐模型相比于 GRU 的各种叠加模型而言,效果更优。

**关 键 词:**电影推荐;用户行为序列;时间权值;GRU

**中图分类号:**TP391.3

**文献标志码:**A

**doi:**10.16836/j.cnki.jcuit.2022.03.001

## 0 引言

行为反映兴趣,通过对用户历史观影行为的研究,可以推断出用户在未来将会对哪些电影感兴趣。因此,研究用户的历史行为序列对电影推荐有着关键的作用。

近年来,随着深度学习技术的快速崛起,研究者们开始使用深度学习技术进行基于用户行为序列的推荐。由于循环神经网络 RNN (recurrent neural network)<sup>[1]</sup>对序列关系有着较强的建模能力<sup>[2-3]</sup>,因此被广泛应用于序列推荐任务<sup>[4-6]</sup>。但因 RNN 只具有短期记忆的能力,所以在面对长序列数据时,经常会出现梯度消失的情况。于是,GRU(gated recurrent units)<sup>[7]</sup>随之出现并应用于推荐系统<sup>[8]</sup>,其核心思路是在网络中加一些门控单元来控制信息的流入和流出,目的是去掉其中的冗余信息,使梯度传播变得更加容易,因此具有长期记忆的能力。

为了有效地从用户的历史行为序列中提取出用户在每个行为下潜藏的兴趣状态,文中将使用 GRU 模型来学习序列中的信息。然而在实际的推荐应用中,随着时间的推移,用户的兴趣会发生相应的变化,而愈接近当前时刻的历史行为对当前时刻的兴趣状态的影响愈大。GRU 只能通过历史行为推断出用户在某个时刻下的兴趣状态,并未考虑到历史行为对兴趣状态的影响存在随时间衰减的这一特征性。因此,提出了 TGRU (time-weighted GRU) 模型,该模型将时间权值与 GRU 模型相融合,以此获得更加精确的兴趣表达。

此外,用户最终兴趣状态的形成与历史的兴趣状态息息相关。因此,在获得用户的所有历史兴趣状态后,学习这些兴趣状态的演化过程对于最终兴趣状态的预测也十分关键。虽然使用 GRU 进行序列学习的模型取得了较好的效果,但是这些方法没有考虑到序列中单个项目对当前决策的重要性,因此基于注意力机制<sup>[9]</sup>的神经网络模型被提出并应用于序列推荐任务中<sup>[10-11]</sup>。DIEN (deep interest evolution network)<sup>[12]</sup>模型中的 AUGRU (GRU with attentional update gate) 单元将注意力机制和 GRU 网络进行融合,捕捉与最终兴趣状态相关的那部分历史兴趣状态的演化过程。因此,将 AUGRU 模型用于学习兴趣的动态变化性是十分可行的。

综上,提出了一种新的融合时间权值和用户行为序列的电影推荐模型——MRTUB (movie recommendation based on time weights and user behavior sequences)。该模型首先使用 TGRU 改进模型提取用户的兴趣状态,也就是对哪些电影感兴趣,感兴趣程度是多少;然后使用 AUGRU 模型捕捉与最终兴趣状态相关的那些历史兴趣状态的演化过程,从而预测用户未来的兴趣。

## 1 相关技术

### 1.1 GRU 模型

GRU 是 LSTM 网络的一种变体。GRU 将 LSTM 网络的 3 种门控装置修改为更新门和重置门这 2 种门控装置。前者决定需要传递多少信息给下一个节点,

其值愈大说明要传递的信息就愈多;后者决定前一个节点的状态信息需要忽略多少,其值愈大说明要忽略的信息愈少。GRU 神经元结构如图 1 所示。

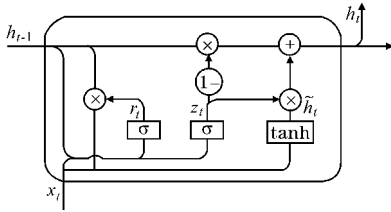


图 1 GRU 神经元结构图

其计算过程如下:

$$z_k = \sigma(W_z x_k + U_z h_{k-1} + b_z)$$

$$r_k = \sigma(W_r x_k + U_r h_{k-1} + b_r)$$

$$\tilde{h}_k = \tanh(W_h x_k + r_k \odot U_h h_{k-1} + b_h)$$

$$h_k = (1 - z_k) \odot h_{k-1} + z_k \odot \tilde{h}_k$$

其中,  $z_k$  表示第  $k$  层更新门的输出;  $r_k$  表示第  $k$  层重置门的输出;  $\tilde{h}_k$  表示候选隐藏状态;  $h_k$  表示输出隐藏状态;  $W_z, W_r, W_h, U_z, U_r, U_h, b_z, b_r, b_h$  都是学习参数。输入向量  $x_k$  和上一个 GRU 单元输出的隐藏状态  $h_{k-1}$  先分别进行线性变换, 然后两者再相加, 最后使用 sigmoid 函数进行激活, 得到的就是更新门的结果。重置门  $r_k$  与前一个 GRU 单元输出的隐藏状态  $h_{k-1}$  做点积运算, 再和输入向量  $x_k$  分别进行线性变换, 然后两者相加, 最后经过一个 tanh 激活函数, 得到的就是重置门的结果。最终, 前一个 GRU 单元输出的隐藏状态  $h_{k-1}$ 、当前节点的候选隐藏状态  $\tilde{h}_k$  和当前节点的更新门  $z_k$  共同决定了  $h_k$  的输出值。

## 1.2 注意力机制

注意力机制通过为序列中的每项分配概率权重, 使重点信息突显出来, 得到充分的关注, 进而提高模型的性能。其结构如图 2 所示。

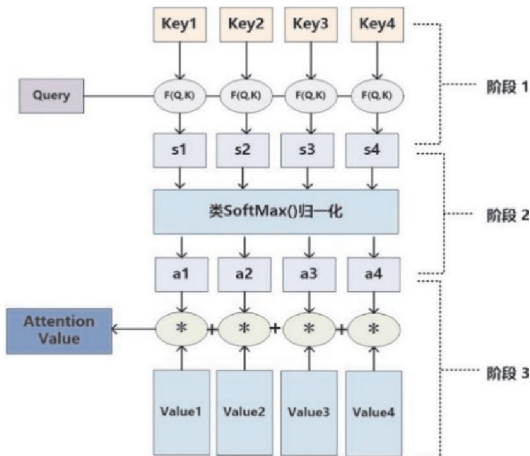


图 2 注意力机制计算过程

注意力机制有 3 个输入: Query、Key 和 Value。Query 表示查询向量, 需要与 Key 向量进行匹配操作; Key 表示键向量, 需要进行被匹配的操作; Value 代表值向量, 一般取 Value = Key。注意力机制的计算过程大致可以分为以下三步:

第一步: 计算 Query 向量和 Key 向量这两者的相关程度, 常用的计算方法有点积法、相似性法和感知机法。

(1) 点积法。将 Query 和 Key 进行点积计算, 得到的相关权重为

$$s_i = k_i^T q$$

其中,  $k_i$  表示序列中的第  $i$  个 Key 向量;  $q$  表示 Query 向量;  $s_i$  表示向量  $k_i$  和向量  $q$  的相关权重。

(2) 相似性法。Cosine 函数可以计算 Query 和 Key 之间的相似性, 这两者之间的相关权重为

$$s_i = \frac{k_i \cdot q}{\|k_i\| \cdot \|q\|}$$

(3) 感知机法。将 Query 向量和 Key 向量作为多层神经网络的输入, 两者之间的相关权重为

$$s_i = W_2 \tanh(W_1 \text{Concat}(k_i, q) + b_1) + b_2$$

其中,  $W_1, W_2, b_1$  和  $b_2$  都是学习参数;  $\text{Concat}(\cdot)$  表示拼接函数。

第二步: 将求得的相关权重进行归一化处理, 归一化操作一般由 SoftMax 函数完成, 归一化后的相关权重为

$$a_i = \frac{\exp(s_i)}{\sum_{j=1}^n \exp(s_j)}$$

第三步: 将  $a_i$  与  $v_i$  进行加权相乘再求和, 即可得到注意力机制的最终输出结果:

$$\text{AttentionScore} = \sum_{i=1}^n (a_i \cdot v_i)$$

其中,  $v_i$  表示第  $i$  个 Value 向量;  $a_i$  为该向量对应的权重系数。

## 1.3 AUGRU 模型

一般情况下, 用户的兴趣是动态变化的。如果能捕捉到与最终兴趣状态相关的那些历史兴趣状态的演化过程, 通过学习该演化过程, 就可以推断出用户未来的兴趣。但并不是每个历史兴趣状态都对最终的兴趣状态有影响, 这里只需要相关的那部分信息。至于相关性程度, 就由注意力得分所决定。DIEN 模型提供了一个捕捉用户兴趣动态变化的方案——AUGRU, 该模型将注意力机制与 GRU 模型相结合, 精确地捕捉到与最终兴趣状态相关的那部分兴趣的演化过程。AUGRU 模型的基本结构如图 3 所示。

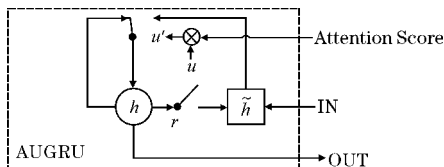


图3 AUGRU 模型结构图

它的原理与 GRU 十分相似,不同之处在于,它将注意力机制嵌入到整个模型中。先计算序列中的每个兴趣状态与最终兴趣状态的相关性程度,即注意力得分。再用注意力得分乘以更新门,将得到的新的更新门替换原始的更新门,其他部分与 GRU 无异。

AUGRU 能够让与用户最终兴趣状态相关的兴趣合理演化而不受非相关兴趣的影响,同时有效地捕获序列之间的前后依赖。因此,文中在电影推荐模型的研究中借鉴应用了该模型。

## 2 一种基于 GRU 的兴趣提取改进模型——TGRU 模型

在预测用户是否会对某部电影进行点击观看之前,需要了解该用户的兴趣,因为兴趣往往会影响行为。所以,提取用户的兴趣十分关键。如上所述,GRU 是序列推荐的典型模型,利用 GRU 模型可以学习用户的历史行为序列,从而提取用户的兴趣。然而,用户的兴趣会随着时间变化而变化,愈接近当前时刻的历史行为对当前时刻的兴趣状态的影响愈大。GRU 并未考虑到历史行为对当前兴趣状态的影响存在随着时间推移衰减的这个现象,所以,文中提出了一种引入时间权值的 GRU 改进模型——TGRU。该模型的核心思想是,距离当前兴趣点越远的历史行为,对当前兴趣点的影响越小,相应地,其时间权值就越大,时间权值为

$$n_k^t = \frac{1}{1 + \alpha(T - t)} \quad t \leq T, k \leq K$$

其中,时间  $T$  代表序列上最后一个用户行为发生的时间; $K$  代表行为序列的长度; $\alpha$  是衰减系数,对于兴趣变化越快的用户, $\alpha$  取值越大; $n_k^t$  表示序列上  $t$  时刻发生的第  $k$  个行为对应的时间权重。

在此基础上,将时间权值融入 GRU 模型便可进行更加精确的兴趣提取。文中提出了 3 种时间权值的融合方案。

方案一:GRU 有一个当前的输入  $x_k$ ,和上一个节点传递下来的隐藏状态  $h_{k-1}$ ,这个隐藏状态包含了之前节点的相关信息。结合  $x_k$  和  $h_{k-1}$ ,经过更新门和重置门的运算操作,GRU 会得到当前隐藏节点的输出  $h_k$ 。本方案的融合方式就是直接将时间权值  $n_k^t$  与第  $k$  个 GRU 单元输出的隐藏状态  $h_k$  相乘,得到一个新的

输出  $h'_k$ :

$$h'_k = h_k \cdot n_k^t$$

方案二:GRU 的更新门决定需要传递多少信息给下一个节点,其值愈大说明要传递的信息就愈多。该思想同样可应用于历史行为的时间权值对于用户最终兴趣状态的影响,时间权值越大,说明此刻的行为距离序列上最后一个行为的时间间隔越小,表示该时刻的信息状态可带入更多至下一个节点。因此,可将时间权值替换掉更新门,具体的实现如下:

$$r_k = \sigma(W_r x_k + U_r h_{k-1} + b_r)$$

$$\tilde{h}_k = \tanh(W_h x_k + r_k \odot U_h h_{k-1} + b_h)$$

$$h_k = (1 - n_k^t) \odot h_{k-1} + n_k^t \odot \tilde{h}_k$$

由此可以看出,与 GRU 不同的是,本方案取消了更新门  $z_k$ ,直接用时间权值  $n_k^t$  进行替换,得到一个新的输出  $h_k$ 。

方案三:本方案的核心思想与方案二相似,但本方案更加保守,并不是直接替换更新门,而是先将更新门与时间权值相乘,再将这个新的更新门替换原始的更新门,其他部分保持不变,它的实现如下:

$$z_k = \sigma(W_z x_k + U_z h_{k-1} + b_z)$$

$$z'_k = n_k^t \cdot z_k$$

$$r_k = \sigma(W_r x_k + U_r h_{k-1} + b_r)$$

$$\tilde{h}_k = \tanh(W_h x_k + r_k \odot U_h h_{k-1} + b_h)$$

$$h_k = (1 - z'_k) \odot h_{k-1} + z'_k \odot \tilde{h}_k$$

通过对比实验得出,方案三相比于方案一、方案二和原始的 GRU,其推荐准确率更优。因此,选取方案三作为最终的时间权值融合方案。限于篇幅,对比实验内容略去。

## 3 一种新的电影推荐模型——MRTUB

### 3.1 模型整体框架

MRTUB 首先将用户历史行为序列作为输入数据;然后使用一个嵌入矩阵对高维且稀疏的向量进行嵌入操作,使之成为低维且稠密的向量;接着,使用提出的 TGRU 改进模型学习用户的历史行为序列,从中提取用户在每个行为下透露出的兴趣状态;接下来,使用 AUGRU 模型捕捉与最终兴趣状态相关的那些历史兴趣状态的演化过程;最后,经过 MLP 完成最终的兴趣预测。MRTUB 推荐模型架构主要分为 5 层,分别是输入层、嵌入层、兴趣提取层、兴趣演化层和输出层,其模型结构如图 4 所示。



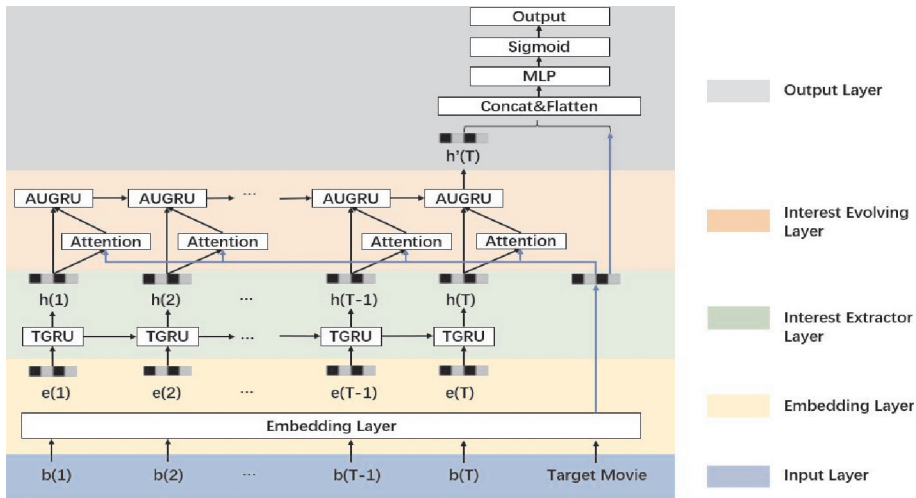


图 4 MRTUB 模型结构图

3.2 输入层

文中所爬取的豆瓣用户历史观影行为数据将作为输入数据,这些数据已经按照时间降序进行了排序。将输入层的用户行为序列长度固定为  $K$ ,这个  $K$  是所有用户序列中最长序列的长度,长度不够  $K$  时就使用填充项进行填充。每一个行为特征包括电影特征  $it$  和反馈信息  $fb$ (用户对电影的评分等级),其定义为

$$b = (it, fb)$$

因为每个用户的打分标准不同,比如有些用户打分比较严格,4 分就意味着对电影非常满意了,而有些用户打分比较宽松,达不到非常满意的程度也会打 5 分。所以,需要对用户的评分进行归一化处理:

$$fb = \frac{score_i - score_{min}}{score_{max} - score_{min}}$$

其中,  $score_i$  表示用户对行为序列中第  $i$  部电影的评分;  $score_{max}$  和  $score_{min}$  分别是用户的最高评分和最低评分。

那么输入的用户行为序列可以表示为

$$S = [b_1, \dots, b_k, \dots, b_K]$$

其中,  $S$  是用户历史行为序列的特征集合;  $b_k$  表示序列中第  $k$  个行为的特征。

3.3 嵌入层

嵌入层的功能是将数据向量化,传统的向量表示多使用 one-hot 编码的形式,但这种方式得到的向量是高维且稀疏的,会对计算机的数据处理能力形成巨大挑战。为此,文中使用嵌入矩阵将 one-hot 编码的高维且稀疏的向量嵌入成低维且稠密的向量,然后再输入模型中,大大地减少了计算机的计算量。

每个行为特征  $b_k$  都需要与一个嵌入矩阵  $M \in \mathbb{R}^{n \times d}$  相乘,再得到嵌入后的向量。其嵌入过程为

$$e_k^{(it)} = M^T b_k^{(it)}$$

$$e_k^{(fb)} = M^T b_k^{(fb)}$$

其中,  $b_k^{(it)}$  和  $b_k^{(fb)}$  分别表示  $b_k$  特征中的电影特征  $it$  和反馈信息  $fb$  的 one-hot 编码;  $e_k^{(it)} \in \mathbb{R}^d$  和  $e_k^{(fb)} \in \mathbb{R}^d$  是经过嵌入之后的特征向量表示。

3.4 兴趣提取层

在这一层,使用提出的 TGRU 改进模型从用户的历史行为序列中挖掘用户潜藏的兴趣状态。每一个用户行为都有一个时间权值  $n_k^t$ ,代表时刻  $t$  发生的第  $k$  个历史行为在时间轴上对用户最终的兴趣状态的影响程度。一个 TGRU 单元的结构如图 5 所示。

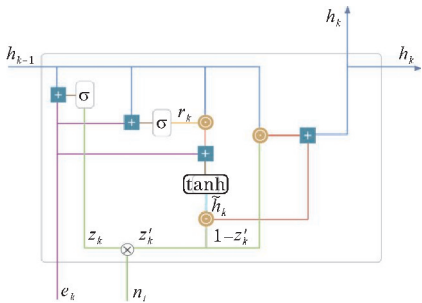


图 5 TGRU 单元结构图

其具体的计算过程如下:

(1) 计算更新门  $z_k$ :

$$z_k = \sigma(W_z e_k^{(it)} + U_z h_{k-1} + b_z)$$

其中,  $e_k^{(it)}$  为输入向量,代表用户第  $k$  个历史行为的电影特征的嵌入向量;  $h_{k-1}$  为上一个 TGRU 单元输出的隐藏状态;  $W_z, U_z, b_z$  是可以学习的参数。

(2) 计算新的更新门  $z'_k$ :

$$z'_k = z_k \cdot n_k^t$$

新的更新门由于加上了时间权值,因此能更加准确地传递所需的信息,同时还能减少梯度消失的风险。

(3) 计算重置门  $r_k$ :

$$r_k = \sigma(W_r e_k^{(it)} + U_r h_{k-1} + b_r)$$

重置门用于决定前一个节点的状态信息需要忽略多少,计算公式与更新门一致,但是这两者的用处和参数不相同。

(4) 计算当前的记忆内容  $\tilde{h}_k$ :

$$\tilde{h}_k = \tanh(W_h e_k^{(it)} + r_k \odot U_h h_{k-1} + b_h)$$

重置门的计算结果是一个 0 到 1 组成的向量,值的大小可以衡量门控开启的大小。如果计算出来的门控值为 0,就代表对应元素的信息将会被完全忽略掉。

(5) 计算最终的记忆内容  $h_k$ :

$$h_k = (1 - z'_k) \odot h_{k-1} + z'_k \odot \tilde{h}_k$$

这一步确定了最终要保存的信息,包括前一个时间步要保存的信息和当前时间步要记忆的信息两部分,并输出传递至下一个 TGRU 单元。

经过兴趣提取层后,用户在每个行为下的兴趣状态就被提取出来了,最终得到序列  $\tilde{S} = [h_1, \dots, h_k, \dots, h_K]$ 。

### 3.5 兴趣演化层

兴趣演化层将使用 AUGRU 模型<sup>[13]</sup>来捕捉与最终兴趣相关的那部分兴趣的演化过程。因为 AUGRU 模型融合了注意力机制,所以需要先计算序列  $\tilde{S}$  中每个隐藏向量的注意力得分。选取了点积法来计算相关权重:

$$a_j^{(it)} = \frac{\exp(e^{it:Query}(h_j^{it:Key})^T)}{\sum_{k=1}^K \exp(e^{it:Query}(h_k^{it:Key})^T)}$$

其中,  $e^{it:Query}$  表示候选电影的嵌入向量;  $h_k^{it:Key}$  表示兴趣提取层输出的第  $j$  个隐藏状态;  $a_j^{(it)}$  得分越高,则代表关系越紧密。

同时,为了更加准确地表示用户兴趣,本层将融合用户评分特征。由于每个用户的打分习惯不同,需要对每个用户的评分进行归一化处理,然后嵌入为低维向量,得到  $e_k^{(fb)} \in \mathbb{R}^d$ 。再将  $e_k^{(fb)}$  和  $h_k^{(it)}$  进行 Concat,作为 AUGRU 模型的输入向量。

AUGRU 的主要计算过程如下:

$$e_k^{(fb,it)} = \text{Concat}(e_k^{(fb)}, h_k^{(it)})$$

$$u_k = \sigma(W_u e_k^{(fb,it)} + U_u h_{k-1} + b_u)$$

$$r_k = \sigma(W_r e_k^{(fb,it)} + U_r h_{k-1} + b_r)$$

$$u'_k = a_k^{(it)} \cdot u_k$$

$$\tilde{h}_k = \tanh(W_h e_k^{(fb,it)} + r_k \odot U_h h_{k-1} + b_h)$$

$$h_k = (1 - u'_k) \odot h_{k-1} + u'_k \odot \tilde{h}_k$$

其中,  $h_k^{(it)}$  为兴趣提取层输出的第  $k$  个隐藏状态;  $a_k^{(it)}$  为  $h_k^{(it)}$  对应的注意力得分;  $W$ 、 $U$ 、 $b$  均为可训练参数。最后一个 AUGRU 单元输出的  $h_K$  将作为兴趣演化层的输出向量进行后面的运算。

### 3.6 输出层

输出层主要将  $h_K$  向量经过 MLP 传递后最终输出预测值完成整个 MRTUB 模型的传播过程,如下:

$$e = \text{Concat}(e_a, h_K)$$

$$x = \text{ReLU}(W_2 \text{ReLU}(W_1 e + b_1) + b_2)$$

$$\text{output} = \text{Sigmoid}(W_3 x + b_3)$$

其中,  $e_a$  为候选电影的嵌入向量; output 为最终的预测结果。

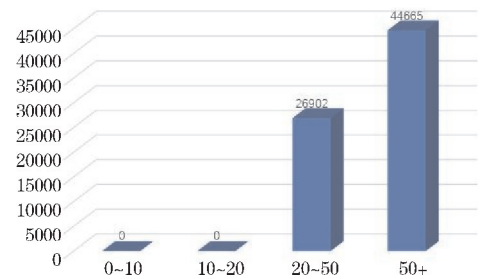
## 4 模型实验

### 4.1 实验数据集

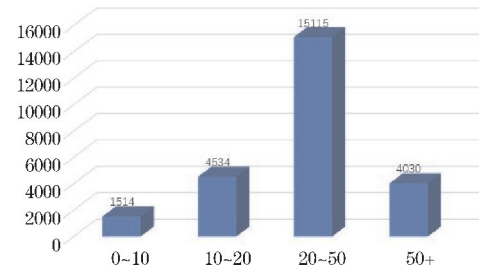
实验使用了一个公开的数据集 ML-10 m 和一个本文通过爬取得到的豆瓣用户观影记录数据集 Douban-HisWatch。这两个数据集的统计信息如表 1 所示。

表 1 数据集统计表

name	users	movies	behaviors
ML-10m	71,567	10,681	10,000,054
Douban-HisWatch	25,193	49,102	1,083,092



(a) ML-10m



(b) Douban-HisWatch

图 6 数据集序列长度情况统计

数据集的稀疏程度会影响一个模型的表现,本文通过在两个稀疏程度不同的数据集上进行实验,来验

证不同推荐模型的性能,这两个数据集的统计情况如图 6 所示。由图 6 可知,Douban-HisWatch 数据集中超过 80% 的用户的行为序列长度小于 50,用户行为较为稀疏;ML-10 m数据集超过 60% 的用户的行为序列长度大于 50,用户行为相较于 Douban-HisWatch 数据集来说非常稠密。

4.2 参数设计

MRTUB 模型在实验过程中获得的最佳参数组合如表 2 所示。

表 2 MRTUB 模型参数设置

参数名	参数值
优化函数 optimizer	Adam
学习率 lr	0.001
Dropout	0.2
batch_size	300
序列长度	100
Embedding_size	64
GRU 隐藏层	128
Attention_size	128

4.3 实验结果与分析

实验将 MRTUB 模型和 GRU+GRU、TGRU+GRU、DIN<sup>[15]</sup>、GRU+AUGRU 这几个模型进行比较,验证了 TGRU 以及 TGRU+AUGRU 的优势。由于样本是以正负样本对的形式保存,因此从验证集和测试集中截取用户前 80% 的行为来预测用户是否会产生那 20% 的行为。这些模型的对比实验结果如表 3 所示。

表 3 模型对比实验结果

数据集	模型	AUC
Douban-HisWatch	GRU+GRU	0.6340
	TGRU+GRU	0.6387
	DIN	0.6412
	GRU+AUGRU	0.6506
	MRTUB(本文模型)	0.6561
ML-10m	GRU+GRU	0.7605
	TGRU+GRU	0.7607
	DIN	0.7634
	GRU+AUGRU	0.7711
	MRTUB(本文模型)	0.7740

由表 3 可知,文中提出的 MRTUB 模型比其他 5 种模型预测更加准确。简单的 GRU 叠加方式效果最差,因为它没有考虑到单个项目对于最终决策的影响;

使用了注意力机制的 DIN 和 GRU+AUGRU 效果明显提升;而融合了时间权值以及注意力机制的 MRTUB 模型效果最好。

5 结束语

用户的兴趣千变万化,如何准确提取出用户的兴趣并提供相应的推荐服务,给研究者们带来了极大的挑战。文中提出的 MRTUB 模型全面地考虑了用户兴趣的提取及演化过程,不仅可以通过 TGRU 改进模型准确学习到用户在每个行为下透露出的兴趣状态,还能适应用户兴趣的动态变化,给出理想的预测结果。这对于无法快速做出选择的用户来说,无疑是正中下怀。实验结果表明,文中的 MRTUB 模型取得了很好的推荐效果。

参考文献:

[1] T Mikolov, S Kombrink, L Burget, et al. Extensions of Recurrent Neural Network Language Model[ C]. Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, 2011: 5528–5531.

[2] J Manotumruksa, E Yilmaz. Sequential-based Adversarial Optimization for Personalized Top-n Item Recommendation[ C]. Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, 2020: 2045–2048.

[3] P Wang, Y Fan, L Xia, et al. KERL: A Knowledge-guided Reinforcement Learning Model for Sequential Recommendation[ C]. Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, 2020: 209–218.

[4] B Hidasi, A Karatzoglou, L Baltrunas, et al. Session-based Recommendations with Recurrent Neural Networks[ C]. Proceedings of the 4th International Conference on Learning Representations, 2016: 1–10.

[5] B Hidasi, A Karatzoglou. Recurrent Neural Networks with Top-k Gains for Session-based Recommendations[ C]. Proceedings of the 27th ACM International Conference on Information and Knowledge Management, 2018: 843–852.

- [6] 袁涛,牛树梓,李会元. 一种基于 Cw-rnn 的多时间尺度序列建模推荐算法[J]. 中文信息学报, 2020,34(6):97–105.
- [7] Dey R, Salemt F M. Gate-variants of Gated Recurrent Unit (GRU) neural networks [C]. 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS), 2017:1597–1600.
- [8] Xia H, Luo Y, Liu Y. Attention neural collaboration filtering based on GRU for recommender systems [J]. Complex & Intelligent Systems, 2021,7(3):1367–1379.
- [9] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need [C]. Advances in Neural Information Processing Systems, 2017:5998–6008.
- [10] Kang W C, McAuley J. Self-attentive sequential recommendation [C]. 2018 IEEE International Conference on Data Mining (ICDM). IEEE, 2018:197–206.
- [11] Xu C, Feng J, Zhao P, et al. Long- and short-term self-attention network for sequential recommendation [J]. Neurocomputing, 2021,423:580–589.
- [12] Zhou G, Mou N, Fan Y, et al. Deep interest evolution network for click-through rate prediction [C]. Proceedings of the AAAI conference on artificial intelligence, 2019,33(1):5941–5948.
- [13] Zhou G, Zhu X, Song C, et al. Deep interest network for click-through rate prediction [C]. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018:1059–1068.

## A Movie Recommendation Model based on Time Weights and User Behavior Sequences

CHENG Si, TAO Hongcai

(School of Computing and Artificial Intelligence, Southwest Jiaotong University, Chengdu 611756, China)

**Abstract:** The research on the users' historical movie viewing behaviors can provide some opinions and references for movie recommendation. In the practical recommendation application, with the passage of time, the interests of users will change accordingly, and the closer the historical behavior to the current moment, the greater the influence on the interest state of the current moment. However, the existing GRU model can only infer the final interest through historical behaviors, without taking into account the characteristic that the influence of historical behavior on the final interest decays with time. The improved TGRU model first proposed in this paper combines the time weight with the GRU model to obtain a more accurate expression of interest. Furthermore, the AUGRU model is used to capture the evolution process of those interests related to the final interest, thereby predicting the users' future interest. Based on the above results and analysis, this paper finally proposes a movie recommendation model, i. e., MRTUB, which integrates time weights and user behavior sequences. The experimental results show that the MRTUB movie recommendation model proposed in this paper has better effect than various superposition models of GRU.

**Keywords:** movie recommendation; user behavior sequence; time weight; GRU