

# 基于用户多类型兴趣波动趋势预测分析的个性化推荐方法\*

丁 浩 李树青

(南京财经大学信息工程学院 南京 210023)

**摘要:**【目的】构建一种基于用户多类型兴趣波动特征预测的推荐方法以提升推荐效果。【方法】针对每种类型用户评分数据加入时间衰减因子并使用神经网络对数据波动线性拟合,选择最优参数结果并对比评估方法有效性。【结果】通过 5 组不同的用户数据进行仿真实验,结果表明,本文方法预测结果的 MAE 和 RMSE 分别较对比方法最高降低幅度达到 47.63%和 44.61%。【局限】由于时间波动的分析依赖用户历史数据,当历史数据量过于稀疏时需采用额外冷启动算法对数据进行预处理。【结论】该方法结合用户对不同商品类型兴趣漂移特征的波动分析和预测,使推荐结果更准确。

**关键词:** 时间序列 兴趣类型 波动分析 个性化推荐

分类号: TP391

DOI: 10.11925/infotech.2096-3467.2019.0370

## 1 引言

随着互联网的快速发展,信息过载成为互联网信息发展的难题之一。推荐系统的出现有效缓解了这一问题,其中应用最广泛的协同过滤推荐算法,其核心思想是通过建立用户和信息间的联系过滤掉冗余无关信息,将更符合用户兴趣的内容提供给用户<sup>[1]</sup>。在实际情况中,由于用户和项目数量庞大种类繁多,传统协同过滤存在计算量过大和数据稀疏性等问题,为解决这一问题,很多学者对传统的协同过滤算法进行改进<sup>[2]</sup>,在算法中引入用户(产品)之间的隐含关系<sup>[3]</sup>,但此类方法忽略了用户偏好可能会随着时间动态变化的特点,这会导致推荐效果随时间推移而不稳定。

本文通过分析用户对不同类型商品的兴趣波动特征,利用神经网络训练用户对不同商品类型的兴趣变化趋势的特征模型,并提取特征参数对样本数据进行预测。本文主要的贡献有:

(1) 利用用户对不同商品类型兴趣变化的波动特征预测用户兴趣变化趋势,用线性方法描述兴趣波动性变化的延伸性和有向性。

(2) 结合用户兴趣漂移特征的特点对用户历史兴趣影响衰减,使用神经网络训练最优特征系数。

(3) 无需繁杂的社会网络信息,建模训练所需信息更易获取,利用线性关系预测可降低时间复杂度,进一步增强系统响应性能,提升用户满意度。

## 2 相关工作

### 2.1 基于概率依赖的时序预测模型

基于概率的时序模型主要是利用随机过程中事件概率分布依赖对用户行为预测,应用较广的有马尔可夫模型,Liu 等利用用户行为的周期性特征处理用户行为并改进马尔可夫状态计算方法,对电影分级数据集进行时序推荐<sup>[4]</sup>。Rendle 等引入基于马尔可夫链的个性化转移矩阵以捕捉时间信息并解决转移矩阵的稀疏

通讯作者: 李树青, ORCID: 0000-0001-9814-5766, E-mail: leeshuqing@163.com。

\*本文系江苏省研究生科研与实践创新计划项目“大数据场景下用户兴趣模式演变趋势分析”(项目编号: KYCX18\_1391)和国家社会科学基金项目“基于大数据分析的数字图书馆个性化服务模式创新研究”(项目编号: 16BTQ030)的研究成果之一。

问题<sup>[5]</sup>。此类方法可以建立随机事件的预测模型,但更侧重于现在状态对未来时刻的影响,而与过去的历史无直接关系,不利于中长期预测。

### 2.2 基于隐式反馈的时序预测模型

隐式反馈为不直接表现用户倾向的历史数据,是推荐系统信息来源的重要部分<sup>[6]</sup>。隐式反馈信息量比显示反馈大很多,反馈来源于用户的自然表现,更接近于真实指标。在现有推荐系统中运用最为广泛的为矩阵分解法<sup>[7]</sup>,基于时间因素的矩阵分解系统则是将时间因素与现有模型结合,通过对时间变化特征的学习进一步提高推荐准确率<sup>[8]</sup>。Koren 提出 TimeSVD++ 算法,将时间信息加入到用户和产品的特征向量中,有效解决兴趣漂移问题<sup>[9]</sup>。除此之外,因式分解方法也被用于隐式反馈的时序建模。Hosseini 等构建了循环泊松因式分解(Recurrent Poisson Factorization, RPF)框架,该框架利用泊松过程对隐式反馈进行建模,RPF 模型将时间视为模型的自然组成部分,并将一系列丰富的时间敏感因子构建时序行为模型,在相应的时间向用户推荐<sup>[10]</sup>。隐式反馈建模方法时间复杂度较高,在实际运用中有很大的限制。

### 2.3 基于神经网络的时序预测模型

近年来,神经网络算法已经广泛运用于图像处理<sup>[11]</sup>、语音识别<sup>[12-13]</sup>、文本分类等方面<sup>[14-15]</sup>,除此之外神经网络也成功运用在时序预测方面<sup>[16-17]</sup>。Yu 等提出一种新的基于递归神经网络的动态循环模型(Dynamic REcurrent bAsket Model, DREAM),该方法引入排名框架(Bayesian Personalized Ranking, BPR)作为目标函数,捕捉用户在不同时间的动态兴趣特征,有利于提高推荐效果<sup>[18]</sup>。使用神经网络的优点在于数据利用率高,可以充分整合不同维度数据,数据处理方式灵活,可以采用分布式计算等方法增强运算吞吐量。其缺点是信息维度较少时训练精确度较低。

上述相关研究工作都有不同的改进,但仍有许多不足,例如:已有算法时间或空间复杂度较高,数据获取难度大,缺乏对时序的有向性和延伸性研究,未充分考虑历史数据对用户的影响等。本文基于时序建

立用户偏好预测模型,结合不同的商品类别的兴趣波动特征,考虑用户历史行为对未来的影响,对于有向的兴趣特征进行建模,利用兴趣趋势的延伸性预测用户兴趣趋势,并使用神经网络算法优化训练预测模型,据此提出一种高效的推荐方法。

## 3 主要工作

### 3.1 问题描述

用户在选择商品前会通过各种途径了解商品信息,所收集的信息会对其选购行为产生影响。随着时间推进,用户根据购买经验逐步建立自己的兴趣类型评判标准,对商品所包含的某种或多种类型保持稳定的兴趣变化水平。比如,不喜欢恐怖类型电影的用户可能因信息中透露出恐怖相关的信息而给出较低评分,并且随着时间推移,用户兴趣关注类型的电影评分可能随着偏好改变而发生变化。本文重点研究商品类型与用户兴趣变化波动的趋势,结合用户兴趣时序下波动特征预测未来数据。

### 3.2 时间片数据模型

为了进一步研究用户对不同商品类型偏好度与时间变化的关系,需先对商品类型进行划分。用户偏好的类型可能是单类型也可能是多重类型的组合,所以考虑构建用户基于类型评分的数据模型。商品通常被划分为多种单一类型,为确定用户感兴趣的类型集合,将每个用户  $U_i$  对应每种单类型或者多种单类型的组合看作一种用户偏好类型,记为  $G_j$ ,将包含偏好类型的商品评分  $R_n$  所构成的评分向量表示如公式(1)所示。

$$\mathbf{R}(U_i, G_j) = (R_1, R_2, \dots, R_n)^T \quad (1)$$

用户选购行为可看作按时序发生的事件向量,为研究用户对商品兴趣趋势变化的情况,将每个用户的类型评分按时序划分,将用户  $U_i$  的某种商品类型  $G_j$  中按时序划分的  $m$  个评分数据作为一个时序数据片段  $S_{j,k}$ ,按照时序方向重新排列每一种类型的评分,基于时间片划分的类型评分矩阵  $\mathbf{I}'_{RU_i}$  如公式(2)所示。其中,  $\mathbf{I}'_{RU_i} \times \max(k_1, k_2, \dots, k_n), n \in \mathbb{N}$ 。

$$\mathbf{I}'_{RU_i} = \begin{bmatrix} (GR_{1,1} \cdots GR_{1,m})_{S_{1,1}} & (GR_{1,m+1} \cdots GR_{1,2m})_{S_{1,2}} & \cdots & (GR_{1,(k_1-1)m+1} \cdots GR_{1,n_1})_{S_{1,k_1}} \\ (GR_{2,1} \cdots GR_{2,m})_{S_{2,1}} & (GR_{2,m+1} \cdots GR_{2,2m})_{S_{2,2}} & \cdots & (GR_{2,(k_2-1)m+1} \cdots GR_{2,n_2})_{S_{2,k_2}} \\ \vdots & \vdots & \ddots & \vdots \\ (GR_{j,1} \cdots GR_{j,m})_{S_{j,1}} & (GR_{j,m+1} \cdots GR_{j,2m})_{S_{j,2}} & \cdots & (GR_{j,(k_n-1)m+1} \cdots GR_{j,n_m})_{S_{j,k_n}} \end{bmatrix} \quad (2)$$

### 3.3 时序类型偏好特征构建

#### (1) 用户行为趋势分析

通过分析用户每种类型的时序序列找出其演变趋势特征  $F^G(t) = [f_1(t), f_2(t), \dots, f_{|F|}(t)]^T$ , 利用最优特征值进行趋势延展性变化拟合并进行预测。为进一步挖掘用户兴趣变化趋势, 使用实验中的数据集中分析用户对不同类型的评分数据方差随时间推移所呈现的变化趋势, 判断数据的收敛程度。

按每个用户评分的时间戳分别使用年、月、日三种时间尺度将数据集划分为时间数据片段, 发现在三种时序尺度下分别有 79.28%、79.04%、73.95% 的用户某些类型的时间数据片段中的评分方差随时间推移而缩小, 通过数据分析发现评分发生收敛的用户在时间维度上对某些类型评分变化趋向于逐步稳定, 用户兴趣点也会随时间推移逐渐落在特定的类型范围内。

#### (2) 趋势分析方法

进一步利用线性关系拟合用户总体趋势变化, 基本思想是利用最小二乘原理使实际数据  $\mu_t$  与预测结果  $\tilde{\mu}_t$  的误差  $E_t$  的平方和达到最小, 参考文献[19], 计算方法如公式(3)所示。

$$\left(\sum_{t=1}^n E_t^2\right)_{\min} = \left(\sum_{t=1}^n (\mu_t - \tilde{\mu}_t)^2\right)_{\min} \quad (3)$$

为描述变量与预测结果的关系, 本文构建趋势特征方程, 如公式(4)所示。

$$\tilde{\mu}_t = \sigma \cdot W(S^G(t), \theta) + \delta + \nabla'_t \quad (4)$$

其中,  $\sigma$  和  $\delta$  是待定系数, 取决于所满足最小值的条件条件极值;  $\theta$  是时间片长度;  $\nabla'_t$  是用户在  $t$  时刻兴趣变化的偏置; 函数  $W$  是变换函数, 主要是将时间片段内的评分数据提取数据均值。用户对于商品的兴趣随着时间推移不断变化, 因此考虑预测数据与实际数据误差的同时也考虑时间对误差的影响, 将误差按照时间远近赋予权重, 从预测精度的角度来说, 在具有稳定趋势变化的数据中, 离当前时间越近的时间片数据误差比越久远的误差更重要, 按照时序将由远及近的评分误差和分别赋予权重因子  $\varepsilon^0, \varepsilon^1, \varepsilon^2, \dots, \varepsilon^{|S|-1}$ , 误差和<sup>[19]</sup>计算如公式(5)所示。在公式(5)中加入权重后展开得到公式(6)。

$$L = \sum_{t=1}^n (\mu_t - \tilde{\mu}_t)^2 \quad (5)$$

$$L_Q = \varepsilon_{|S|-1}^{n-1} (\mu_{n-1} - \tilde{\mu}_{n-1})^2 + \varepsilon_{|S|-2}^{n-2} (\mu_{n-2} - \tilde{\mu}_{n-2})^2 + \dots + \varepsilon_2^1 (\mu_1 - \tilde{\mu}_1)^2 + \varepsilon_1^0 (\mu_0 - \tilde{\mu}_0)^2 \quad (6)$$

基于文献[19]所提误差计算方法, 结合本文所提趋势特征方程, 将函数  $\tilde{\mu}_t$  与公式(4)分别代入公式(6), 化简后如公式(7)所示。

$$L'_Q = \sum_{t=1}^n \varepsilon_{|S|-t+1}^{n-t} (\mu_t - \sigma \cdot W(S^G(t), \theta) - \delta - \nabla'_t)^2 \quad (7)$$

其中,  $0 < \varepsilon < 1$ , 显然最近的误差  $\varepsilon^0 = 1$ , 表明最近的误差有最高的权重。根据时间的远近权重因子  $\varepsilon$  按比例递减, 且因子衰减的速度取决于  $\varepsilon$  值,  $\varepsilon$  越接近 0 衰减越快, 越接近 1 衰减越慢。

为使  $L'_Q$  达到最小, 使用多元函数极值法对待定系数  $\sigma$  和  $\delta$  求偏导数, 求导过程如公式(8)和公式(9)所示。

$$\frac{\partial L'_Q}{\partial \delta} = -2 \sum_{t=1}^n \varepsilon_{|S|-t+1}^{n-t} \cdot \mu_t + 2\delta \sum_{t=1}^n \varepsilon_{|S|-t+1}^{n-t} + 2\sigma \sum_{t=1}^n \varepsilon_{|S|-t+1}^{n-t} \cdot W(S^G(t), \theta) + \Delta_{bias}^\delta \quad (8)$$

$$\frac{\partial L'_Q}{\partial \sigma} = -2 \sum_{t=1}^n \varepsilon_{|S|-t+1}^{n-t} \cdot \mu_t \cdot W(S^G(t), \theta) + 2\delta \sum_{t=1}^n \varepsilon_{|S|-t+1}^{n-t} \cdot W(S^G(t), \theta) + 2\sigma \sum_{t=1}^n \varepsilon_{|S|-t+1}^{n-t} \cdot W^2(S^G(t), \theta) + \Delta_{bias}^\sigma \quad (9)$$

其中,  $\Delta_{bias}^\delta$  和  $\Delta_{bias}^\sigma$  是用户兴趣偏置在不同参数下求导的偏移量, 联立公式(8)和公式(9), 令偏导数  $\frac{\partial L'_Q}{\partial \delta} = \frac{\partial L'_Q}{\partial \sigma} = 0$ , 解得待定系数  $\sigma$  和  $\delta$  的值, 可使得  $L'_Q$  取得最小值, 此时  $\sigma$  和  $\delta$  的值为最优兴趣特征系数。

### 3.4 基于多类型趋势的神经网络推荐框架

#### (1) 训练方法

在模型框架中, 将转化好的数据分为若干段, 再通过时间片评分方差的收敛程度判定数据趋势特征是否满足稳定的延伸趋势, 如果满足则使用趋势预测模型进行训练, 方法是通过初始化数据预测下一时间点用户可能给出的数据, 并计算与真值的误差, 在上一一次结果的基础加入新的训练数据进行预测和验证, 并通过调整用户时序兴趣偏置, 最终使趋势的预测模型与实际数据的误差和最小; 如果用户所选购商品中不含有趋势延伸特性的类型, 则使用协同过滤方法进行预测。

#### (2) 目标函数及预测方法

实验训练模型分为输入转化、训练、转化输出三个部分, 如图 1 所示。①-③属于输入转化模块, ④-⑤属于训练模块, ⑥-⑦属于转化输出模块。

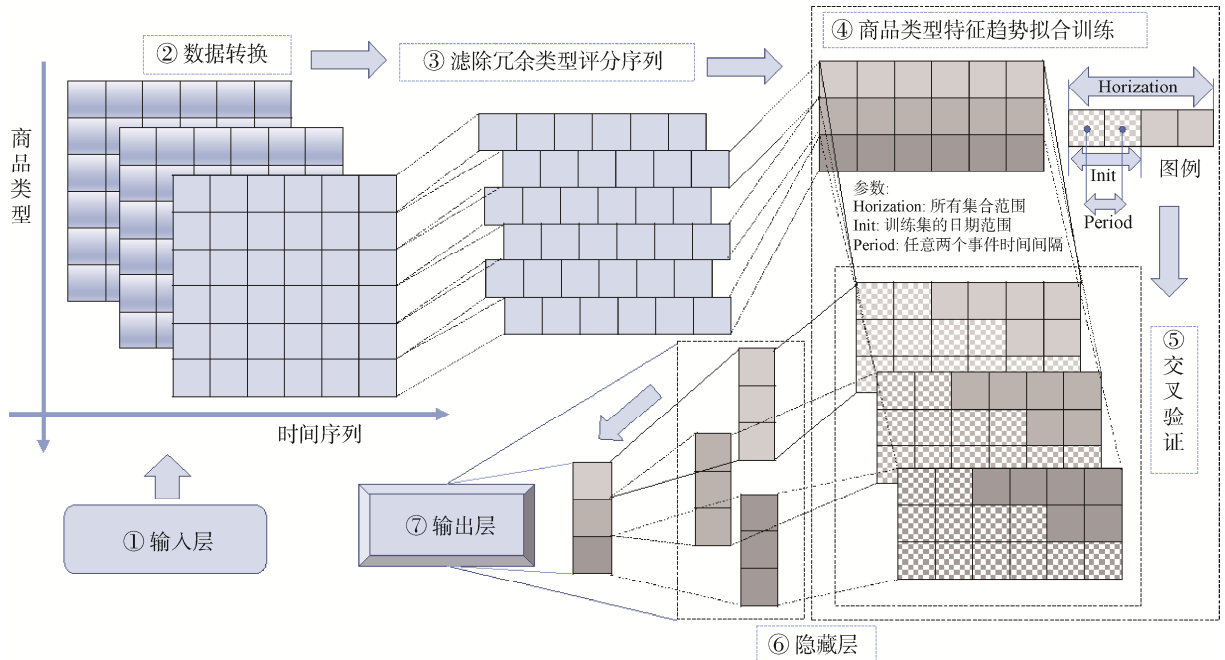


图1 模型框架示意图

将用户对不同商品类型评分从集合  $G$  中按时间顺序取出, 分别放入集合  $V_{u_i} = \{S_1, S_2, \dots, S_{k_n}\}$ , 作为框架的输入层(Input Layer)。输入层的数据传入数据转换模块, 在转换模块中进行扁平化处理, 将数据通过类型分类, 并按照时间序列以二维矩阵的形式存储。转化后的数据通过步骤③进行预筛选处理, 对数据初步分析, 找出可能会影响拟合训练精度的数据, 剔除冗余噪声数据, 将符合条件的数据重新按分类、按时序预测, 在隐藏层中按照选定的超参  $\theta$ -时间片长度, 根据每种类型进行拟合训练。本文通过线性关系构建预测函数, 如公式(10)所示。

$$\varphi(x^{(t)}, \theta) = \theta_0 + \theta_1 x_1^{(t)} \quad (10)$$

其中,  $\theta_0$  为偏移量,  $\theta_1$  为系数,  $x^{(t)}$  表示  $t$  时刻样本点,  $\varphi(x^{(t)}, \theta)$  表示估计值, 因为超参  $\theta$ -时间片长度的设置影响实验框架整体输出, 因此在后续实验中将继续讨论其设置。为统一矩阵化便于计算, 给每个坐标点  $x$  增加一维, 如公式(11)所示。

$$(x_1^{(t)}, y^{(t)}) \rightarrow (x_0^{(t)}, x_1^{(t)}, y^{(t)}), \forall x_0^{(t)} = 1 \quad (11)$$

联立公式(10)和公式(11)则有:

$$\varphi(x^{(t)}, \theta) \rightarrow \theta^T x \quad (12)$$

为找出最优兴趣特征系数, 根据文献[19-20], 使用误差和构建目标函数, 在此基础上加入时间衰减因

子  $w^{(t)}$ , 同时防止过拟合在目标函数中加入正则化项, 正则化处理后的目标函数如公式(13)所示。

$$L(\theta) = \frac{1}{2k} \sum_{t=1}^k w^{(t)} (y^{(t)} - \varphi(x^{(t)}, \theta))^2 + \frac{\lambda}{2} \sum_{i=1}^n \theta_i^2 \quad (13)$$

其中,  $k$  表示时间样本点数量,  $y^{(t)}$  是  $t$  时刻用户数据的真值,  $\theta$  为待优化兴趣特征系数,  $\lambda$  是正则参数,  $\sum_{i=1}^n \theta_i^2$  是正则项, 目标函数中常数项  $\theta_0$  并未正则化, 因此  $\lambda$  不宜设置过大, 否则会导致所有参数过小而因变量接近常数出现欠拟合。  $w^{(t)}$  表示  $t$  时刻误差项的权重, 权重因子  $\varepsilon$  是一组范围在  $[0,1]$  的常数, 根据时序由远到近衰减速度由快到慢衰减误差项, 本文提出表示方式如公式(14)所示。

$$w^{(t)} \rightarrow \varepsilon^{k-t} \quad (14)$$

因所求目标函数是加权误差和, 为得到使误差最小的系数  $\theta$ , 根据文献[20]的优化方法选择随机梯度下降法(Stochastic Gradient Descent, SGD)对目标函数进行优化。对公式(13)求导, 过程如公式(15)所示。

$$\frac{\partial L(\theta)}{\partial \theta} = \frac{w^{(t)}}{k} (y^{(t)} - \varphi(x^{(t)}, \theta)) \cdot \frac{\partial}{\partial \theta} \left( \sum_{t=1}^k (\theta^T x^{(t)} - y^{(t)}) \right) + \frac{\lambda \theta}{2} \cdot \frac{\partial}{\partial \theta} \sum_{i=1}^n \theta_i^2 \propto \frac{w^{(t)}}{k} (y^{(t)} - \varphi(x^{(t)}, \theta)) x_{\theta}^{(t)} + \lambda \theta \sum_{i=1}^n \theta_i \quad (15)$$



进而使用随机梯度下降法优化, 对所有样本点每一次迭代沿梯度的反方向更新参数, 直至收敛, 如公式(16)所示。

$$\Theta \leftarrow \Theta + \alpha \left( \frac{1}{k} \sum_{t=1}^k w^{(t)} (y^{(t)} - \varphi(x^{(t)}, \Theta)) x_{\Theta}^{(t)} + \lambda_{\Theta} \sum_{i=1}^n \Theta_i \right) \quad (16)$$

其中,  $\alpha$  为学习因子, 用于控制学习速率, 对于训练过程仅作为特征提取, 不设置激活函数。在训练过程中利用数据训练每个用户对不同商品类型的兴趣特征系数, 每种类型选择误差和小于阈值的  $m$  组兴趣特征系数构建预测模型, 其中误差和的阈值取对应类型的所有误差总和均值。对于满足条件的兴趣特征系数, 用于预测用户相应类型的评分, 假设用户当前选购评分行为是  $t$  时刻, 下一次评分行为即将发生在  $t+1$  时刻,  $t+1$  时刻用户  $U$  对包含类型  $G$  的项目评分预测方法如公式(17)所示。

$$\text{Predict}R(U_G^{t+1}) = \frac{\sum_{j=1}^{n_G} \left( \frac{\sum_{i=1}^m (\Theta_0^{(i)} + \Theta_1^{(i)} x_{\Theta_i}^{(t)})}{m} \times |x_{j_G}^{(t)}| \right)}{\sum_{j=1}^{n_G} |x_{j_G}^{(t)}|} \quad (17)$$

$$= \frac{\sum_{j=1}^{n_G} \left( \frac{\sum_{i=1}^m \Theta_i^T x_{\Theta_i}^{(t)}}{m} \times |x_{j_G}^{(t)}| \right)}{\sum_{j=1}^{n_G} |x_{j_G}^{(t)}|}$$

其中,  $\Theta$  表示符合条件的兴趣特征系数,  $x_{\Theta_i}^{(t)}$  表示在某刻用户对  $G$  类型的评分,  $n_G$  表示包含类型  $G$  的样本集合总数量,  $|x_{j_G}^{(t)}|$  表示包含类型  $G$  的样本集合内样本数量。

### (3) 模型优化

当用户评分的商品类型包含所有的初始类型时, 会产生大量复合类型, 虽然在实际情况中不可能覆盖全部初始类型, 但随着复合类型的元素增多, 组合的长度也随之增加, 复合类型的评分数量趋于稀疏, 同时复合类型过于冗长还易导致训练集数据过拟合, 即使用训练集以外的数据测试时精确度较低; 另一方面, 大量的冗余组合数据会导致运算效率严重下降。针对这些问题需要对矩阵  $I'_{RU_i}$  的数据进行预优化处理, 由于在完全划分的情况下包含评分数量少于两个的复合类型数量较多, 因此可将此类复合类型与其子集合并, 从而改善预测效果。

## 4 实验

### 4.1 数据集和评价方法

MovieLens 数据集是美国明尼苏达大学 GroupLens 项目组公开的电影评分数据集, 包含从 IMDB、The Movie Data Base 上得到的用户对电影的评分信息。本文使用的数据是 MovieLens 20M 数据集, 其数据规模如表 1 所示。

表 1 数据集规模

用户数量	电影数量	评分数量	稀疏度(%)
138 493	27 278	20 000 263	0.5294

本实验中 Dropout 值设为 0.5, 误差正则化系数  $\gamma$  设置为 0.0002, 学习率  $\alpha$  设置为 0.001, 训练迭代次数分别设为 5、10、15 次。

为更准确地评估本文的推荐算法, 使用数据集里的用户模拟在线用户, 用户与用户之间相互独立, 随机抽取 5 000 名用户进行实验, 将其平均分为 5 组, 每组之间互不相交。首先对用户的数据进行预处理, 把每个评分里电影对应的初始 Genre 集合作为新一列附加在对应评分后, 然后将每个用户数据按照时间戳排序, 将每个用户的评分按照评分时间先后顺序的前 75% 作为训练集, 后 25% 作为测试集, 训练集和测试集类型划分如表 2 所示, 划分完数据之后, 每个测试集的训练集和测试集数据约按照 3:1 分布, 满足测试的需求。

参考文献[21]的评价方法, 使用平均绝对误差 (Mean Absolute Error, MAE) 和均方根误差 (Root Mean Squared Error, RMSE) 作为评价指标。假设使用测试集生成预测评分  $\{p^{(1)}, p^{(2)}, \dots, p^{(i)}\}$ , 相对应的实际评分  $R(x^{(i)})$ , MAE 和 RMSE 的计算方法分别如公式(18)和公式(19)所示。

$$MAE = \frac{1}{N} \sum_{i=1}^N |p^{(i)} - R(x^{(i)})| \quad (18)$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (p^{(i)} - R(x^{(i)}))^2} \quad (19)$$

这两种误差值越小, 表示算法的预测准确度越高。

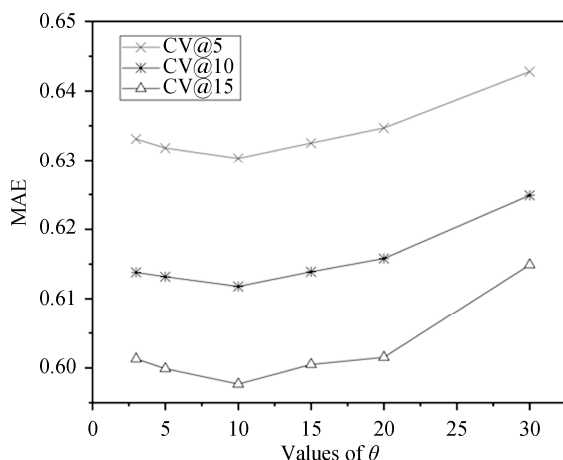
表 2 测试数据划分标签数量

Genre/Set	Set#1		Set#2		Set#3		Set#4		Set#5		Total	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
Action	31 344	11 022	30 990	11 147	27 434	9 686	30 954	10 891	30 661	10 924	151 383	53 670
Adventure	24 134	8 561	24 306	8 679	21 295	7 528	23 980	8 611	24 704	8 763	118 419	42 142
Animation	5 576	2 559	5 661	2 533	5 206	2 275	5 707	2 520	6 046	2 722	28 196	12 609
Children	9 317	2 945	9 180	2 988	8 220	2 602	8 907	2 819	10 115	3 122	45 739	14 476
Comedy	42 591	14 162	42 343	13 892	37 222	12 000	40 770	13 269	44 719	14 557	207 645	67 880
Crime	18 343	6 071	18 627	6 178	16 251	5 478	17 945	5 953	18 248	6 207	89 414	29 887
Documentary	906	800	897	799	760	750	875	786	941	848	4 379	3 983
Drama	48 023	16 793	49 602	17 172	42 774	15 218	47 443	16 580	49 783	17 723	237 625	83 486
Fantasy	11 197	4 472	11 210	4 498	9 873	3 881	10 979	4 357	11 578	4 562	54 837	21 770
Film-Noir	1 274	276	1 223	324	1 063	276	1 278	306	1 324	316	6 162	1 498
Horror	9 156	3 095	7 674	2 619	7 544	2 552	7 678	2 655	8 451	2 707	40 503	13 628
IMAX	2 175	1 424	2 276	1 485	1 825	1 268	2 130	1 374	2 391	1 491	10 797	7 042
Musical	4 828	1 560	5 025	1 609	4 298	1 443	4 814	1 440	5 256	1 710	24 221	7 762
Mystery	8 895	2 961	8 707	2 929	7 734	2 624	8 494	2 827	8 817	2 966	42 647	14 307
Romance	21 445	6 704	22 305	7 008	18 970	5 936	21 486	6 452	23 009	7 135	107 215	33 235
Sci-Fi	18 223	6 129	17 597	5 988	15 823	5 342	17 444	6 001	17 545	5 949	86 632	29 409
Thriller	30 174	10 228	29 597	10 126	26 384	9 126	28 868	9 893	29 582	10 154	144 605	49 527
War	5 959	1 732	6 025	1 836	5 119	1 520	6 049	1 816	6 251	1 855	29 403	8 759
Western	2 608	893	2 357	805	2 085	693	2 360	798	2 399	870	11 809	4 059
Total	296 168	102 387	295 602	102 615	259 880	90 198	288 161	99 348	301 820	104 581	1 441 631	499 129

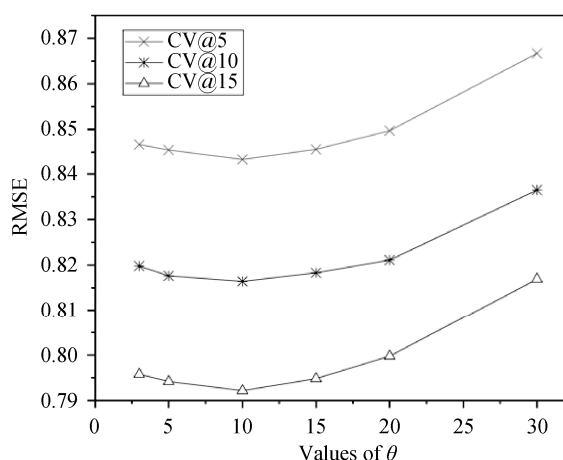
### 4.2 参数影响

在本文所提 TGE (Time Genres Embedded)推荐算法中,  $\theta$  参数的大小代表时间数据片段长度, 过长的时间片段会造成片段内样本数量过于稀疏, 导致趋势延伸准确度下降; 而过短的时间片会造成时间片内数据量过少, 造成时间片的特征值不精确, 同时也会形成过多的冗余数据, 因此时间片长度取值会对预测结果有较大影响, 为探究  $\theta$  值与推荐效果之间的关系, 实验分别按照不同训练迭代次数进行测试比较。

参数  $\theta$  变化对评价指标 MAE 和 RMSE 的影响如图 2 所示, 可以看出参数  $\theta$  对实验结果影响较为明显。当  $\theta$  取值不同时, 两种评价指标的结果也都发生了明显变化。为保证每个时间片内有足够的评分数量计算方差收敛度, 参数  $\theta$  的初始值从 3 开始增长。随着  $\theta$  增长, MAE 和 RMSE 的值逐渐减小。当  $\theta$  达到某个阈值之后(本实验阈值为 10), 两指标的值开始逐渐增加。推荐效果随着参数  $\theta$  变化而变化的情况, 反映出时间片长度所包含信息数量以及时间片的数量对推荐结果产生的影响, 决定推荐效果的好坏。



(a) 参数  $\theta$  对 MAE 的影响



(b) 参数  $\theta$  对 RMSE 的影响

图 2 每种迭代参数  $\theta$  对预测的影响

### 4.3 预测结果

本文选取 4 种经典推荐算法与 TGE 的评估结果进行比较:

- (1) NormalPredictor: 假设训练集是正态分布情况下随机预测评分。
- (2) KNNBasic: 一种基础的协同过滤算法。
- (3) SVD++: 在 SVD 算法基础上考虑用户隐式评分的扩展。

(4) NMF: 一种基于非负矩阵分解的协同过滤算法。

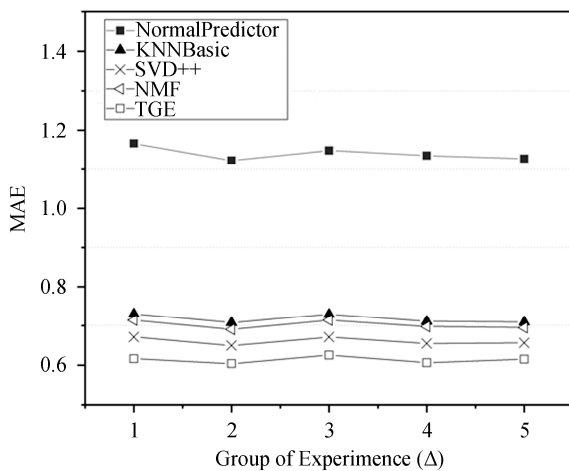
对于 TGE 的交叉验证, 初始数据 Init 给定训练集的 50%, 再分别按迭代次数去将训练集平均划分为若干有序数据段  $X_1, X_2, \dots, X_{|cv|}$ , 并将有序数据段分为训练段和测试段, 每迭代一次再从上一轮的测试段中抽出一段数据  $X_k$  按时序放入训练段, 直至训练段数据完全被使用。最终各个算法的实验结果如表 3 所示。

表 3 不同迭代参数和分组的各种算法 MAE 和 RMSE 比较结果

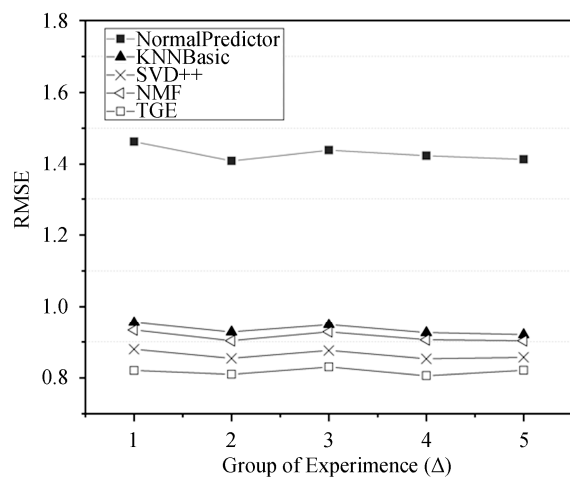
方法	参数	Group of Experience ( $\Delta$ )										Mean( $\Sigma\Delta$ )	
		$\Delta=1$		$\Delta=2$		$\Delta=3$		$\Delta=4$		$\Delta=5$		MAE	RMSE
		MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE		
NormalPredictor	CV@5	1.1673	1.463	1.1224	1.4084	1.1498	1.44	1.1348	1.422	1.1265	1.4117	1.14016	1.429
	CV@10	1.1666	1.461	1.1241	1.4106	1.149	1.4385	1.1373	1.426	1.1273	1.4124	1.14086	1.4297
	CV@15	1.1689	1.464	1.1225	<b>1.4083</b>	1.1476	1.4386	1.1349	1.423	1.1301	1.417	1.1408	1.4302
KNNBasic	CV@5	0.7346	0.96	0.7118	0.9326	0.7336	0.9537	0.715	0.93	0.7132	0.9244	0.72164	0.9401
	CV@10	0.7288	0.953	0.7064	0.9256	0.7283	0.9465	0.71	0.925	0.7085	0.9190	0.7164	0.9337
	CV@15	0.7272	0.951	<b>0.7049</b>	0.924	0.7265	0.9449	0.7082	0.922	0.7069	<b>0.9173</b>	0.71474	0.9319
SVD++	CV@5	0.6753	0.884	0.6533	0.8589	0.675	0.8807	0.6581	0.857	0.6601	0.8608	0.66436	0.8683
	CV@10	0.6706	0.879	0.6481	0.8529	0.6692	0.8742	0.6532	0.852	0.6558	0.8562	0.65938	0.8629
	CV@15	0.6681	0.875	<b>0.6465</b>	0.8507	0.6687	0.873	0.6519	<b>0.850</b>	0.6533	0.8536	0.6577	0.8604
NMF	CV@5	0.7173	0.939	0.6938	0.9074	0.7195	0.9346	0.7023	0.911	0.6986	0.908	0.7063	0.9199
	CV@10	0.7140	0.932	0.6892	0.9017	0.7119	0.9244	0.6966	0.905	0.6932	0.9005	0.70098	0.9127
	CV@15	0.7102	0.928	<b>0.6878</b>	<b>0.8998</b>	0.7115	0.924	0.6944	0.902	0.6929	0.9001	0.69936	0.9108
TGE-CF	CV@5	0.6341	0.852	<b>0.6233</b>	<b>0.8354</b>	0.6334	0.8426	0.6289	0.840	0.6317	0.8472	0.63028	0.8434
	CV@10	0.6127	0.814	0.601	0.8115	0.6279	0.8393	<b>0.5997</b>	<b>0.796</b>	0.6176	0.8214	0.61178	0.8164
	CV@15	0.6023	0.795	<b>0.5862</b>	0.7822	0.6146	0.8081	0.5895	<b>0.782</b>	0.596	0.7941	0.59772	0.7922

不同分组的测试结果比较如图 3 所示。TGE 算法与几种经典的算法相比有较大提高, 即使在迭代次数最少的情况下, 效果仍好于其他算法, 充分说明了本文所提用户时序下商品类型兴趣变化关系的合理性和

有效性。此外, TGE 推荐算法需要的电影类型信息和时间戳序列在实际场景中比较容易集中获取, 不需要复杂的社交网络关系信息或是难以获取的用户标签信息, 所以此方法在实际场景中应用更为广泛。



(a) 不同分组的 MAE 比较结果



(b) 不同分组的 RMSE 比较结果

图 3 不同分组的测试结果比较

#### 4.4 时间评估

针对各种算法每次迭代和响应的具体时间做比较, 实验运行环境为 Intel Core i5 CPU, 主频为 3.4GHz, 8GB 的运行内存, Windows 7 系统, 将 5 次迭代的时间平均值作为平均每次迭代的时间, 响应时间指迭代更新后到程序完成结果计算一共消耗的时间, 最终结果如表 4 所示。

表 4 每次迭代的运行时间和响应时间

评估方法	运行时间(s)	响应时间(s)
NormalPredictor	0.15	0.18
KNNBasic	0.50	2.79
SVD++	558.98	9.81
NMF	7.04	0.19
TGE-CF	1.15	0.15

从表 4 可知, SVD++ 迭代和响应时间最长, 远超过其他算法, 这是因为 SVD++ 算法在矩阵的填充和分解过程消耗了较多时间, 在实际场景中随着数据增多会消耗大量时间, 显然该算法时间效率太低。其次耗时较长的是基于非负矩阵分解的 NMF 算法, 由此可以看出, 在迭代过程中考虑关系越复杂则时间复杂度越高。NormalPredictor 算法假设训练集是正态分布进行随机评估, 没有考虑用户之间和商品之间的关系, 对算法结果的精度影响很大。KNNBasic 只考虑邻居关系, 每次迭代更新后查询相似的邻居会消耗很多时间, 速度取决于用户邻居和评分数量。TGE 算法根据用户偏好类型时序波动变化趋势迭代预测, 在保证推荐效果的同时, 迭代时间在可接受范围且响应时间最短, 在实际应用中能够快速响应用户的请求。

## 5 结 语

针对用户选购商品的行为, 本文提出一种基于时序的用户对多类型兴趣波动趋势预测的推荐算法 TGE, 以用户对类型偏好随时间片变化的特征预测为目标。首先建立基于时间片划分的时序数据模型, 初步抽取时间片特征, 再对满足条件的用户类型兴趣波动趋势建模。通过训练框架提取训练特征, 利用时序类型趋势特征的延伸性预测用户对商品的偏好度。实验结果表明, TGE 算法取得较好的预测效果, 与其他经典算法相比, 时间复杂度也得到改善, 有效提升推荐系统的准确性和用户满意度。

此外 TGE 算法还有一些可改进的方面, 例如对商品类型可以划分得更细, 将用户社交评论与购买行为融合, 结合时序挖掘用户在多种异源信息下更多的偏好特征变化趋势, 以进一步改善推荐效果。

## 参考文献:

- [1] Goldberg D, Nichols D, Oki B M, et al. Using Collaborative Filtering to Weave an Information Tapestry[J]. Communications of the ACM, 1992, 35(12): 61-71.
- [2] Ma H, King I, Lyu M R. Learning to Recommend with Social Trust Ensemble[C]// Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2009: 203-210.
- [3] Konstas I, Stathopoulos V, Jose J M. On Social Networks and Collaborative Recommendation[C]// Proceedings of the 32nd International ACM SIGIR Conference on Research & Development in Information Retrieval. ACM, 2009: 195-202.
- [4] Liu Z, Tan K, Wang X Q, et al. A Learning Framework for Temporal Recommendation Without Explicit Iterative Optimization[J]. Applied Soft Computing, 2018, 67:529-539.
- [5] Rendle S, Freudenthaler C, Schmidt-Thieme L. Factorizing Personalized Markov Chains for Next-Basket Recommendation[C]// Proceedings of the 19th International Conference on World Wide Web. Springer, 2010: 811-820.
- [6] 印鉴, 王智圣, 李琪, 等. 基于大规模隐式反馈的个性化推荐[J]. 软件学报, 2014, 25(9): 1953-1966. (Yin Jian, Wang Zhisheng, Li Qi, et al. Personalized Recommendation Based on Large-Scale Implicit Feedback[J]. Journal of Software, 2014, 25(9): 1953-1966.)
- [7] Jamali M, Ester M. A Matrix Factorization Technique with Trust Propagation for Recommendation in Social Networks[C]// Proceedings of the 2010 ACM Conference on Recommender Systems. ACM, 2010: 135-142.
- [8] Liu J, Wu C, Xiong Y, et al. List-Wise Probabilistic Matrix Factorization for Recommendation[J]. Information Sciences, 2014, 278: 434-447.
- [9] Koren Y. Collaborative Filtering with Temporal Dynamics[J]. Communications of the ACM, 2010, 53(4): 89-97.
- [10] Hosseini S, Alizadeh K, Khodadadi A, et al. Recurrent Poisson Factorization for Temporal Recommendation[C]// Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2017: 847-855.
- [11] Xu Z, Yang Y, Hauptmann A G. A Discriminative CNN Video Representation for Event Detection[C]// Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern



- Recognition. IEEE, 2014: 1798-1807.
- [12] Graves A, Mohamed A R, Hinton G. Speech Recognition with Deep Recurrent Neural Networks[C]// Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2013: 6645-6649.
- [13] Weng C, Yu D, Watanabe S, et al. Recurrent Deep Neural Networks for Robust Speech Recognition[C]// Proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2014: 5532-5536.
- [14] Arevian G, Panchev C. Optimising the Hystereses of a Two Context Layer RNN for Text Classification[C]// Proceedings of the 2007 IEEE International Joint Conference on Neural Networks. IEEE Press, 2007: 2936-2941.
- [15] Wang C, Jiang F, Yang H. A Hybrid Framework for Text Modeling with Convolutional RNN[C]// Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2017: 2061-2070.
- [16] Selvin S, Vinayakumar R, Gopalakrishnan E A, et al. Stock Price Prediction Using LSTM, RNN and CNN-Sliding Window Model[C]// Proceedings of the 2017 International Conference on Advances in Computing. IEEE, 2017: 1643-1647.
- [17] Madan R, Sarathimangipudi P. Predicting Computer Network Traffic: A Time Series Forecasting Approach Using DWT, ARIMA and RNN[C]// Proceedings of the 11th International Conference on Contemporary Computing. IEEE, 2018: 1-5.
- [18] Yu F, Liu Q, Wu S, et al. A Dynamic Recurrent Model for Next Basket Recommendation[C]// Proceedings of the 39th International ACM SIGIR Conference. ACM, 2016: 729-732.
- [19] 田垅, 刘宗田. 最小二乘法分段直线拟合[J]. 计算机科学, 2012, 39(S1): 482-484. (Tian Long, Liu Zongtian. Least-squares Method Piecewise Linear Fitting[J]. Computer Science, 2012, 39(S1): 482-484.)
- [20] Bottou L. Large-Scale Machine Learning with Stochastic Gradient Descent[C]// Proceedings of COMPSTAT'2010. 2010: 177-186.
- [21] Chai T, Draxler R R. Root Mean Square Error (RMSE) or Mean Absolute Error (MAE)? – Arguments Against Avoiding RMSE in the Literature[J]. Geoscientific Model Development, 2014, 7(3): 1247-1250.

### 作者贡献声明:

丁浩: 采集、清洗和分析数据, 进行实验, 论文起草;  
李树青: 提出研究思路, 设计研究方案, 论文最终版本修订。

### 利益冲突声明:

所有作者声明不存在利益冲突关系。

### 支撑数据:

支撑数据由作者自存储, E-mail: 415058975@qq.com。

- [1] 丁浩. 商品类型分类数据.dat. 实验分类数据.  
[2] 丁浩. 时序化评分数据.dat. 实验评分数据.

收稿日期: 2019-04-08  
收修改稿日期: 2019-06-12

## Personalized Recommendation Based on Predictive Analysis of User's Interests

Ding Hao Li Shuqing

(School of Information Engineering, Nanjing University of Finance and Economics, Nanjing 210023, China)

**Abstract:** [Objective] This paper tries to construct a time series prediction model based on the fluctuation of users' historical interests, aiming to improve the recommendation results. [Methods] We added time attenuation factor to the ratings by each type of users and linearly fit the data fluctuation with neural network. Then, we chose the optimal parameters to compare the effectiveness of the proposed method. [Results] We conducted five rounds of user simulation tests and found the MAE and RMSE errors of the proposed method were reduced by 47.63% and 44.61%. [Limitations] Analysis of time fluctuation relies on users' historical data, thus, additional cold-start algorithm is needed to preprocess the data. [Conclusions] The proposed method could effectively analyze and predict the changing of interests in different commodities, and provide more accurate recommendation lists.

**Keywords:** Time Series Interest Type Fluctuation Analysis Personalized Recommendation