

# 基于改进的宽深度模型的推荐方法研究

王艺平<sup>1</sup> 冯旭鹏<sup>2</sup> 刘利军<sup>1</sup> 黄青松<sup>1,3</sup>

<sup>1</sup>(昆明理工大学信息工程与自动化学院 云南 昆明 650500)

<sup>2</sup>(昆明理工大学教育技术与网络中心 云南 昆明 650500)

<sup>3</sup>(云南省计算机技术应用重点实验室 云南 昆明 650500)

**摘要** 现代社交网络的个性化博文推荐中,博文特征选取质量的高低直接影响了推荐的质量和效率。深度模型可以较高质量地提取出文本中句法和语义的特征。然而短文本特征稀疏且未考虑上下文语境的问题,普遍存在于文本推荐任务中。针对以上问题,在现有宽深度模型的基础上,利用门限循环单元对其多层普通神经网络进行改进,提出宽深度门循环联合(Wide & Deep-GRU)模型,进一步探索浅层部分和深度部分的联合训练。使用从新浪微博获取的真实数据集分别与单一逻辑回归模型、单一深度神经网络模型和宽深度模型进行对比。实验表明,该方法整体上推荐质量较高,同时推荐效率较之前模型也有显著提高。

**关键词** 文本推荐 排序模型 深度学习 门循环神经单元

中图分类号 TP3 文献标识码 A DOI:10.3969/j.issn.1000-386x.2018.11.007

## THE RECOMMENDATION METHOD BASED ON IMPROVED WIDE AND DEEP MODEL

Wang Yiping<sup>1</sup> Feng Xupeng<sup>2</sup> Liu Lijun<sup>1</sup> Huang Qingsong<sup>1,3</sup>

<sup>1</sup>(Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, Yunnan, China)

<sup>2</sup>(Educational Technology and Network Center, Kunming University of Science and Technology, Kunming 650500, Yunnan, China)

<sup>3</sup>(Yunnan Key Laboratory of Computer Technology Applications, Kunming 650500, Yunnan, China)

**Abstract** In the personalized blogs recommendation of modern social networks, the quality of blog feature selection directly affects the quality and efficiency of recommendation. Deep model can extract syntactic and semantic features from text in a higher quality. However, the short text features are sparse and do not consider the context, which are ubiquitous in the text recommendation tasks. Aimed at above problem, a wide and deep-GRU model, based on the existing wide and deep model, was proposed by using Gated Recurrent Unit to improve its ordinary multilayer neural networks so as to further explore the joint training of the shallow part and depth part. In this paper, the real datasets obtained from Sina Weibo were respectively compared with single logistic regression model, single depth neural network model and the wide and deep model. Experiments show that this method generally achieves a good recommendation effect and the recommended efficiency is significantly higher than the previous model.

**Keywords** Text recommendation Sorting model Deep learning GRU

## 0 引言

随着在线社交网络系统的盛行,微博为人们提供了一个十分开放的沟通渠道来阅读、评论、引用、社交,其包含了基于文本的微博条目和配置文件、图片、数据

及多媒体等大量有价值的资源<sup>[1]</sup>。面对海量级别的信息,如何快速定位用户特征,如何有效地向用户推荐他们感兴趣的资源和如何基于历史数据探索过去从未或很少发现的特征,利用深度学习技术来改进博文的个性化推荐是一个很好的解决方案。

本文通过对现有推荐模型的限制性进行分析研

究,结合传统机器学习浅层模型与深度模型各自的优点,提出了一种传统浅层线性模型与深度网络模型相结合的推荐排序方法。首先利用基于稀疏自编码器的主题提取方法对博文语料进行主题提取,在百万级的博文数据中取得数百个候选集。接着,在数百个候选集的基础上,一方面获取用户特征等信息,采用交叉特征法处理特征,输入到浅层模型当中。另一方面,只需少量的特征工程,通过映射的方法,处理低维稠密特征得到密集矢量后,输入到以门循环单元为核心的深度模型。最后,处理输出结果并得到博文项的推荐排序。实验表明,本文的方法分别单一逻辑回归模型、单一深度神经网络模型和宽深度模型等进行对比,不仅在短文本数据特征稀疏的情况下使博文推荐的准确率得到了提升,还在一定程度上提高了模型的训练和预测速度,具有更高的推荐效率。

## 1 相关工作

近年来,微博社交网络的个性化推荐服务发展迅速的与其他产品领域相结合,已经经历了一个基本的范式转变,使用深度学习已经几乎成为所有学习问题的通用解决方案<sup>[2]</sup>。本文博文推荐系统的纵览如图1所示,主要的研究工作是图中推荐系统的排序部分。

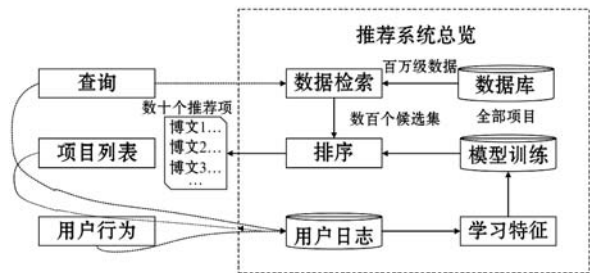


图1 推荐系统总览

与现有工程中大量使用的矩阵分解方法相比<sup>[3]</sup>,较少有研究是使用深度学习方法来推荐服务的。针对传统基于内容的推荐系统中,用户特征稀疏且难以获取等问题,文献[4]通过分析用户的浏览记录和搜索记录,来丰富用户特征等数据<sup>[5]</sup>。文献[6]利用加入评论信息来缓解数据稀疏的问题。文献[7]通过利用用户特征、情境特征和项目特征等多源异构数据,提出了一种同时具有较高的记忆能力和泛化能力的模型,探索过去从未出现或很少发生的新特征组合,从而提升推荐项目的多样性。针对微博博文的短文本特征和容易产生单词稀疏、单词同意等问题,文献[8]提出了一个基于注意力机制的记忆网络来进行博文的推荐。针对文本推荐中的上下文语义理解不明确等问题,文献[9]采用GRU抓住会话中行为之间的依赖关

系,对时间序列上的变化进行建模,输入是具有1-N编码的实际状态,输出为每个项目的下一个会话的可能性。文献[10]为了学习用户的偏好以及更好地理解语义信息,采用利用循环神经网络RNN(Recurrent neural network)来学习用户的隐表示。基于以上研究工作,本文研究发现现有推荐模型的限制和挑战有:

1) 现有传统的推荐方法严重依赖于有效的数据特征提取,通常面临可扩展性不足的问题,使用深度学习方法能够提高特征提取的质量,且适合于大规模数据处理,将大大增强模型的推荐性能。

2) 由于深度模型过强的学习能力,博文中某类出现频率较低的特征难以学习到有效的特征向量表示,存在过于泛化的问题。引入浅层模型如线性模型等可以对出现次数较少的特征进行较好地学习。

3) 现有的深度推荐模型未充分考虑到文本项上下文语义关系与隐表示。优化模型的网络结构,如利用RNN,推荐质量会有较大提升。

4) 博文语料集数据量大,数据维度高,导致用户端的博文推荐效率较低。因而,提升模型的推荐效率十分重要。

## 2 面向博文推荐的宽深度门循环联合模型

本文构建的宽深度门循环联合模型,主要结构组成如图2所示。

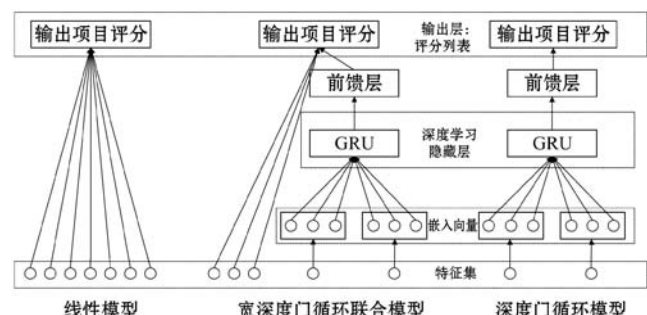


图2 Wide & Deep-GRU 层次结构图

### 2.1 浅层部分

非线性特征变换的广义线性模型被广泛用于稀疏输入的大规模回归、分类问题<sup>[11]</sup>。本文模型的浅层部分采用逻辑回归方法,来预测用户的相关特征与候选博文间能否建立关系,并能对出现次数较少的特征进行较好地学习。通过广义的交叉特征转换,来记忆特征是十分有效且可行的<sup>[7]</sup>。

逻辑回归模型如下:

$$H_w(x) = \frac{1}{1 + e^{-w^T x}} \quad (1)$$

式中:将用户特征集作为逻辑回归分类作为输入,  $\mathbf{x} = [x_1, x_2, \dots, x_d]$  是特征  $d$  的一组向量, 特征集合包括原始输入特征和组合特征。  $\mathbf{w} = [w_1, w_2, \dots, w_d]$  是模型的参数。

本文这样定义交叉特征:

$$\phi_k(x) = \prod_{i=1}^d x_i^{c_{ki}} \quad (2)$$

这里  $c_{ki} \in \{0, 1\}$  是一个布尔值, 如果第  $i$  个特征是第  $k$  个转换  $\phi_k$  的一部分, 那么  $c_{ki}$  即为 1, 否则就是 0。对于二进制特征, 例如, 有且仅当组合特征全部成立才是 1, 否则就是 0。本文利用交叉特征, 不仅获取到了二进制特征之间的交互, 并且将非线性添加到了线性模型当中<sup>[7]</sup>。

## 2.2 深度网络部分

由于现有的深度网络模型的相邻层之间为全连接, 但其单层的各个节点则为无连接, 样本的处理在每个时刻独立, 因而不能对其时间序列上的变化建模<sup>[12]</sup>。门限循环单元 GRU (Gated Recurrent Unit) 是一种更精细的 RNN 单位模型<sup>[12]</sup>, 它将门控制单元从 3 个减少到 2 个, 模型更加简单, 既克服了梯度消失的缺点, 又可以结合上下文语义学习博文的长期依赖信息, 具有更高的效率。

本文的深度门循环部分, 见图 2 右边部分。其核心是 GRU 层, 并且在输出结果层和最后一层之间添加额外的前馈层。将模型隐藏层的节点之间使用门循环单元连接起来, 并控制其隐藏节点的输出, 这样可以有效地建模特征在文本上下文时序的变化<sup>[12]</sup>。门循环单元如图 3 所示。

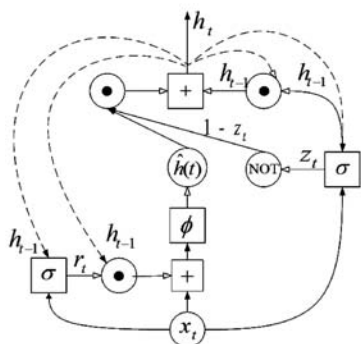


图 3 GRU 内部结构图

模型含多个 GRU 层时, 上一单元层的隐藏状态是下一层的输入。GRU 的激活函数  $h_t$ , 是介于之前激活函数状态和候选激活的线性插值函数  $\hat{h}_t$  之间的, 如下式:

$$h_t = (1 - z_t)h_{t-1} + z_t \hat{h}_t \quad (3)$$

更新门由以下提供:

$$z_t = \sigma(W_z x_t + U_z h_{t-1}) \quad (4)$$

式中:候选激活函数  $\hat{h}_t$  用类似的方式来计算:

$$\hat{h}_t = \tanh(W_x x_t + U(r_t \odot h_{t-1})) \quad (5)$$

最终的重置门由下式给出:

$$r_t = \sigma(W_r x_t + U_r h_{t-1}) \quad (6)$$

每一个高维稀疏的分类特征首先转换成低维实值向量, 本文将其看作是一个映射向量, 映射向量的维度通常是  $O(10)$  到  $O(100)$ 。初始化映射向量, 这些低维映射向量将送到正向通路的 GRU 的隐层中, 权重参数在模型训练的过程中更新, 使得最终的损失函数达到最小<sup>[7]</sup>。实验表明, 单层 GRU 网络性能最优。本文使用  $\tanh$  函数作为输出层的激活函数, 效果明显优于其他函数<sup>[9]</sup>。

## 2.3 宽深度门循环联合模型

联合模型如图 2 中间部分所示。本文组合浅层部分和深度门循环部分, 使用联合输出结果并取对数的加权和作为预测值, 然后将该加权和馈送到一个共同的损失函数进行联合训练。小批量随机优化宽深度门循环联合模型参数, 同时再反向传播到模型的浅层和深度门循环部分。

联合模型的预测函数是:

$$P(Y = 1 | x) = \sigma(W_{\text{wide}}^T [x, \phi(x)] + W_{\text{gru}}^T a^{(l)} + b) \quad (7)$$

式中:  $Y$  是一个二分类标签,  $\sigma(\cdot)$  是 sigmoid 功能函数。  $\phi(x)$  是原始特征  $x$  的交叉特征转换的结果,  $b$  是偏置项。  $W_{\text{wide}}$  是浅层部分的权重向量,  $W_{\text{gru}}$  是应用在最终激活值  $a^{(l)}$  上的权重向量。最终, 将单一候选项得到的概率值处理后, 进行排序得到推荐排序项。

## 2.4 模型训练

在模型训练期间, 输入层接收预处理好的训练数据集。浅层部分的输入特征包括由用户离散特征以及搜索关键词构成的交叉特征。模型的深度门循环部分, 为每个特征学习一个 32 维的嵌入向量, 再将所有特征的嵌入向量与连续特征连接起来, 产生大约为 1 200 维的密集矢量<sup>[7]</sup>。随后, 将连接密集矢量馈送到门循环单元层中, 最终输出为逻辑单元。模型框架如图 4 所示。

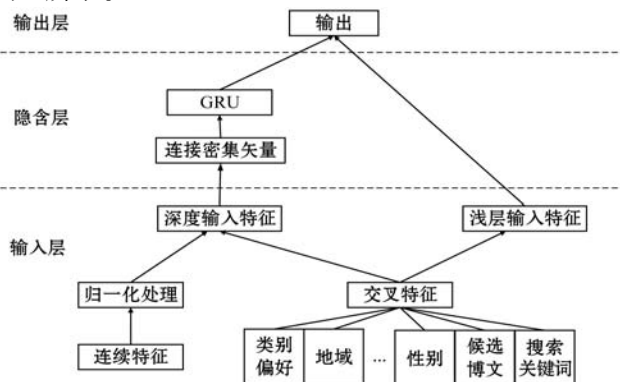


图 4 推荐模型框架

本文的联合训练不同于组合。在一个组合中,单独的模型是分开训练而不知道浅层和深度部分的关联,他们的预测仅在推断的时候组合而不是在训练的时候。然而,本文的联合训练在训练时考虑浅层、深度门循环部分以及其总和的权重,同时优化所有的参数<sup>[7]</sup>。且对于联合训练,浅层部分只需要通过较少的交叉特征来补充深度门循环部分的不足即可。

为了对整个 Wide & Deep-GRU 模型训练,本文定义一个合理的误差损失函数,使用 FTRL (Follow-the-regularized-leader) 算法<sup>[13]</sup> 和 L1 正则化作为模型线性部分的优化器,最终利用 AdaGrad 算法<sup>[14]</sup> 来优化模型的深度门循环部分。对于分类问题,一般采用交叉熵 (cross-entropy) 作为其损失函数。其公式如下:

$$H_{y'}(y) = - \sum_i y'_i \log y_i \quad (8)$$

式中: $y'_i$  为是真实的概率分布 (即训练语料的类别标签),  $y_i$  为模型预测的概率分布。这里通过最小化  $H_{y'}(y)$  的值来训练整个模型。

## 3 实验

### 3.1 实验数据与预处理

本文设计实验来验证 Wide & Deep-GRU 模型推荐性能的高效性和准确性。在数据准备阶段,使用新浪微博官方 API 和 Pyquery 爬虫框架,获取 2017 年 6 月部分真实用户的完整日志数据,进而生成数据集。数据集由用户的微博 ID、属性特征和搜索关键词以及其一个月内发布、转发、评论、赞和收藏的微博等特征信息组成。

本文的实验数据随机选取  $52 \times 10^3$  个用户节点,获得的数据以 json 格式存储。再将预处理过的数据集按 6:2:2 的比例,分别用来做训练数据集,验证数据集以及测试数据集。

本文实验对博文进行了如下预处理:

- (1) 采用停用词字典的方法,来去除停用词等对主题没有贡献的内容。
- (2) 去除微博中的“@用户名”。
- (3) 由于博文长度过短,不能够表现准确的主题信息<sup>[15]</sup>。于是去掉少于 20 字的微博。经处理后的微博共计 4 716 452 条。

### 3.2 深度模型部分超参数设置

本文的模型使用 Google 的开源深度学习框架 TensorFlow 实现。为了兼顾训练速度和模型的性能,模型超参数最终确定为:深度门循环部分将最小批处

理尺寸设置为 20,为每个特征学习一个 32 维的嵌入向量,一个隐藏单元内设置 64 个神经元。将模型训练的迭代次数设置为 200。为了避免模型训练过拟合,在输入层中增加 dropout 操作<sup>[15]</sup>,并将 dropout 丢弃率设置为 0.2。每次取 100 个数据作为一个批次进行训练。

### 3.3 实验评测标准

本文实验推荐性能的评测标准将使用推荐质量和推荐效率来定义。推荐质量使用准确率 (Precision)、召回率 (Recall)、F 值 (F-Measure) 作为衡量指标<sup>[17]</sup>。推荐效率通过对用户进行个性化推荐时,模型的训练和预测速度来衡量<sup>[12]</sup>,即分别在数据训练集和数据预测集中,采用单条博文平均训练时间和预测时间来进行比较,并与其他模型的训练和预测速度比较。

对于用户  $u$ ,  $R_u$  为模型推荐结果的博文集合,  $L_u$  作为用户  $u$  喜欢的博文集合,由此将推荐准确率、召回率和 F 值为:

$$\text{Precision} = \frac{\text{推荐列表中用户 } u \text{ 喜欢的微博数}}{\text{模型推荐结果的微博数}} = \frac{R_u \cap L_u}{R_u} \quad (9)$$

$$\text{Recall} = \frac{\text{推荐列表中用户 } u \text{ 喜欢的微博数}}{\text{用户 } u \text{ 喜欢的微博数}} = \frac{R_u \cap L_u}{L_u} \quad (10)$$

$$\text{F 值} = \frac{2\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (11)$$

本文将用户有操作交互行为的微博,比如用户转发、评论、赞和收藏的微博等都认为是用户喜欢的微博。在实验过程中,宽深度门循环联合模型最终通过迭代式的训练得到。在每一次迭代过程中,模型将根据训练样本集中的数据进行调整,对于本文模型来说,需要成千上万次迭代才能最终收敛。

### 3.4 实验结果对比及分析

本文的实验,通过随机划分不同的训练集,利用包含本文模型在内的 4 种不同的推荐方法,分别进行模型参数的训练学习,在不同的测试集上进行实验,对比验证它们的推荐性能。

下面简要给出其他三种对比方法的介绍:

(1) 单一逻辑回归 (Wide) 模型:将推荐预测问题转为二分类问题,判定一个用户对这条博文点击与否则概率为多少,逻辑函数利用 sigmoid 函数。

(2) 单一深度神经网络 (Deep) 模型:为每个分类特征学习为嵌入向量。再将其馈送到普通深度网络隐层中,最终输出为逻辑单元。

(3) 宽深度 (Wide & Deep) 模型<sup>[7]</sup>:将上述两模型融合,浅层部分的输入特征作为交叉特征,将记忆和

泛化联合起来。

### 3.4.1 推荐质量

本文随机抽取数据集中 200 位用户,其一天中有交互操作行为的全部博文中,只要两条博文之间的时间差大于一小时,本文就认为这两条博文不属于用户的同次浏览行为,遂进行一次推荐,得到所有用户的推荐质量,即结果的准确性、召回率和 F 值。

图 5 和图 6 通过改变不同的参数,一同表明了模型算法的推荐质量,分别展示了随用户人数和推荐列表长度的变化,推荐质量的衡量指标即准确率、召回率和 F 值的改变。由图 5 可以得到,随着用户人数的增多,4 种模型的用户特征更加接近于用户的真实特征,其结果的准确率、召回率和 F 值均呈增长趋势。对于不同大小的训练集,Wide、Deep、Wide & Deep、Wide & Deep-GRU 这 4 种模型方法的最高准确率分别为 0.726、0.722、0.728 和 0.742。随着数据的增多,Wide & Deep-GRU 模型的方法能从这些数据中学到的特征越多,进而越能更加准确的推荐博文<sup>[17-18]</sup>。由图 6 可以得到,随着推荐列表长度的增长,模型准确率下降,召回率在升高,而当推荐列表长度为 10 的时候,F 值最高。即在推荐列表增加到一定长度后,评分高的微博基本都已包含在推荐列表中,推荐列表长度的增加对推荐质量的影响并不会再增加。由图可见,Wide & Deep-GRU 模型综合效果最好。

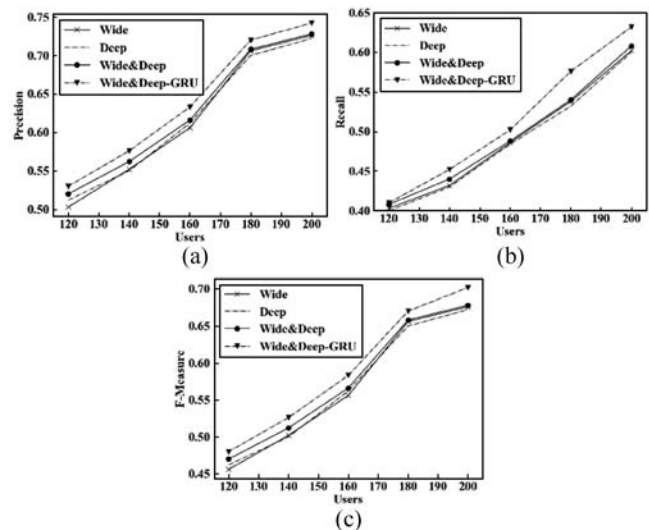


图 5 随用户人数变化的模型推荐质量对比图

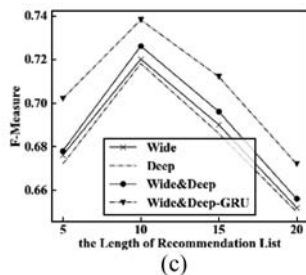
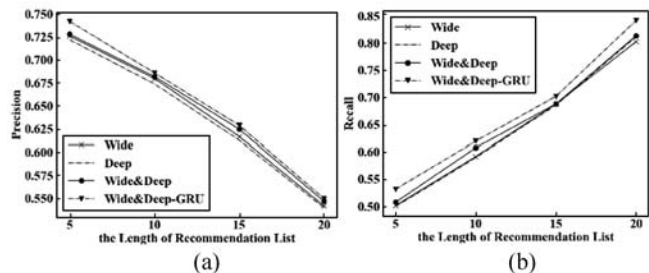


图 6 随列表长度变化的模型推荐质量对比图

### 3.4.2 推荐效率

本文联合模型的推荐效率设计使用推荐过程中的产生单位时间开销来衡量。即分别在数据训练集和数据预测集中,采用单条博文平均训练时间和预测时间来进行比较,并与其他模型的训练和预测速度比较。从表 1 中可以看出,本文提出的 Wide & Deep-GRU 模型,由于其深度模型部分更加简单,克服了梯度消失的缺点,又可以学习博文的长期依赖信息,具有更高的效率,在各个数据集中平均训练时间和预测时间都明显小于 Wide & Deep 模型。具体的,当用户数为 150,推荐长度为 10 时,Wide & Deep-GRU 模型的训练速度平均是 Wide & Deep 模型的 1.917 倍,在测试集上的预测速度平均是 Wide & Deep 模型的 1.686 倍。

表 1 推荐效率对比评估

模型对比	评估指标	
	训练时间/ms	预测时间/ms
Wide & Deep	69	59
Wide & Deep-GRU	36	35

## 4 结 语

本文通过利用深度学习方法构建了宽深度门循环联合模型(Wide & Deep-GRU)模型并对其训练学习、博文推荐,随后分别与其他三种模型进行对比验证,其准确率比传统机器学习方法和现有的深度模型都有较大的提高。利用深度门循环单元学习方法,可以更好地泛化用户属性的特征,改善短文本特征稀疏且未考虑上下文语境等现状。本文仅考虑了离线数据集的评估结果,离线数据集中的印象和标签是固定的,而在线系统可以通过将泛化与记忆混合来产生新的探索性推荐,并从新的用户点击中学习。后期有望进行在线 A/B 测试,实验效果和运算速度等,会有大幅度的提高。本文的浅层模型部分使用了基本的多元线性回归模型,接下来的工作将尝试其他模型结构来代替获得更好的性能。

## 参 考 文 献

- [1] Li H, Ma X P, Shi J, et al. Study on the Method of Microblogging Recommendation Based on the Relationship between Trust and Social Network[J]. Chinese Journal of Information, 2017, 31(2): 146 - 153.
- [2] Covington P, Adams J, Sargin E. Deep Neural Networks for YouTube Recommendations[C]//Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 2016: 191 - 198.
- [3] Su X, Khoshgoftaar T M. A survey of collaborative filtering techniques[J]. Advances in Artificial Intelligence, 2009, 2009(12):421425.
- [4] Elkahky A M, Song Y, He X, et al. 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. 2015: 278 - 288.
- [5] 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/OL]. 计算机学报, 2018 年在线发布[2018 - 02 - 28]. <http://cjci.ict.ac.cn/online/bfpub/hlww-2018124152810.pdf>.
- [6] Zheng L, Noroozi V, Yu P S. Joint deep modeling of users and items using reviews for recommendation[C]//Proceedings of the Tenth ACM International Conference on Web Search and Data Mining. ACM, 2017: 425 - 434.
- [7] Cheng H T, Koc L, Harmsen J, et al. Wide & Deep Learning for Recommender Systems[C]//Proceedings of the 1st Workshop on Deep Learning for Recommender Systems. ACM, 2016: 7 - 10.
- [8] Huang H, Zhang Q, Huang X, et al. Mention Recommendation for Twitter with End-to-end Memory Network[C]//Twenty-Sixth International Joint Conference on Artificial Intelligence. 2017:1872 - 1878.
- [9] Hidasi B, Karatzoglou A, Baltrunas L, et al. Session-based Recommendations with Recurrent Neural Networks[EB]. eprint arXiv:1511.06939, 2015.
- [10] Okura S, Tagami Y, Ono S, et al. Embedding-based News Recommendation for Millions of Users[C]//Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2017:1933 - 1942.
- [11] Xu C Y, Li D, Wang S, et al. Based on blog and network structure information friends recommended method[J]. Journal of Computer Applications and Applications, 2016, 52(1): 55 - 60.
- [12] Li X L, Duan H, Xu M, et al. Chinese segmentation method based on neural network of gate cycle element[J]. Journal of Xiamen University (Natural Science Edition), 2017(2): 237 - 243.
- [13] McMahan H B. Follow-the-Regularized-Leader and Mirror Descent: Equivalence Theorems and L1 Regularization[J]. eprint arXiv:1009.3240, 2011.
- [14] Duchi J, Hazan E, Singer Y. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization[J]. Journal of Machine Learning Research, 2011, 12(7):257 - 269.
- [15] Hinton G E, Srivastava N, Krizhevsky A, et al. Improving neural networks by preventing co-adaptation of feature detectors[EB]. eprint arXiv:1207.0580, 2012.
- [16] Shi L, Tao Y C, Li J Y, et al. Study on real-time recommendation model of personalized microblogging[J]. Small Microcomputer System, 2016, 37(9): 1910 - 1914.
- [17] Hidasi B, Quadrana M, Karatzoglou A, et al. Parallel Recurrent Neural Network Architectures for Feature-rich Session-based Recommendations[C]//Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 2016: 241 - 248.
- [18] Tan Y K, Xu X, Liu Y, et al. Improved Recurrent Neural Networks for Session-based Recommendations[EB]. eprint arXiv:1606.08117, 2016.

## (上接第 41 页)

题,提高开发测试的效率。另外,平台的自动部署功能将实现秒级自动完成系统的部署,提高系统部署及升级的效率,还支持平台容器的组件以及非容器组件的统一部署。这些新特性的引入,将改变原有的开发、测试及部署的流程,因此对于新流程的规范管理和协调将是容器化平台建设的又一重点任务。

## 参 考 文 献

- [1] 刘鹏. 云计算[M]. 3 版. 北京:电子工业出版社, 2015:1 - 9.
- [2] 武志学. 云计算导论:概念 架构与应用[M]. 北京:人民邮电出版社, 2016:43 - 52.
- [3] 顾炯炯. 云计算架构技术与实践[M]. 2 版. 北京:清华大学出版社, 2016:108 - 115.
- [4] Johnston J, Batchelli A, Cormack J, et al. Docker 生产环境实践指南[M]. 北京:人民邮电出版社, 2016:1 - 31.
- [5] 浙江大学 SEL 实验室. Docker 容器与容器云[M]. 2 版. 北京:人民邮电出版社, 2016:221 - 419.
- [6] Bass L. DevOps:软件架构师行动指南[M]. 北京:机械工业出版社, 2017:202 - 219.
- [7] 武志学. 云计算虚拟化技术的发展与趋势[J]. 计算机应用, 2017, 37(4):915 - 923.
- [8] 王亚玲, 李春阳, 崔蔚, 等. 基于 Docker 的 PaaS 平台建设[J]. 计算机系统应用, 2016, 25(3):72 - 77.
- [9] 汪 恺, 张功萱, 周秀敏. 基于荣其虚拟化技术研究[J]. 计算机技术与发展, 2015, 25(8):138 - 141.
- [10] 孙海洪. 微服务架构和容器技术应用[J]. 金融电子化, 2016(5):63 - 64.