

文章编号: 2095-2163(2021)07-0166-06

中图分类号: TP391

文献标志码: A

基于序列特征提取的个性化推荐

陈 昀, 张瑞林

(浙江理工大学 信息学院, 杭州 310018)

摘要: 基于序列的推荐是推荐系统研究的热点, 序列中包含许多重要信息, 如物品的点击规律和用户的兴趣, 有效利用序列信息是提高推荐准确率的关键。为了有效提取序列信息, 提出了 ACRec 推荐系统模型, 利用多头自注意力机制和卷积神经网络从动态和静态两个方面提取序列信息, 并利用矩阵分解增强模型中用户与序列的语义关系。在 MovieLen-1M 和 Video_Games 两大公开数据集上实验证明, 相比于其他基线模型, ACRec 提高了推荐的准确率, Hit@10 分别提高了 1.03% 和 18.4%, NDCG@10 分别提高了 2.6% 和 20.9%。

关键词: 推荐系统; 注意力机制; 卷积神经网络; 矩阵分解

Personalized recommendation based on sequence feature extraction

CHEN Yun, ZHANG Ruilin

(School of Informatics Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China)

[Abstract] Sequential recommendation is a hot topic in the research of recommendation system. Sequences contain a lot of important information, such as item click rules and user interests. Effective use of sequence information is the key to improve the accuracy of recommendation. In order to effectively extract the sequence information, a model of recommendation system called ACRec is proposed, which uses multi-head self-attention mechanism and Convolutional Neural Network to extract sequence information from both dynamic and static aspects, and uses matrix factorization to enhance the semantic relationship between users and sequences in the model. Experiments on the two public data sets of MovieLen-1M and Video_Games prove that compared with other baseline models, ACRec improved the accuracy of recommendation, the Hit@10 increased by 1.03% and 18.4%, and the NDCG@10 increased by 2.6% and 20.9%.

[Key words] recommendation system; attention; Convolutional Neural Network; matrix factorization

0 引言

在互联网时代, 信息过载已经成为不可避免的问题^[1]。推荐系统是解决信息过载问题的重要方法。传统的推荐系统对用户和物品的每一次交互进行建模。但在现实生活中, 用户的下一次交互往往与用户的上一次交互有关。基于序列的推荐系统考虑了用户历史点击序列对下一次点击的影响, 如 Zhou 等人^[2]提出了 DIN 推荐系统模型, 通过注意力模型计算目标物品与序列中物品的相关性, Hidasi 等人^[3]提出了利用 GRU 进行会话推荐。

虽然目前已有许多基于序列的推荐模型, 但还存在一些问题: 首先, 对于序列特征提取只考虑序列的发展变化特征或只考虑序列的静态特征, 没有对 2 方面特征同时进行处理; 其次, 模型只对序列进行计算, 忽略了用户与序列的语义关系。

为了解决上述问题, 本文使用注意力机制和卷积神经网络分别学习序列的动态特征和静态特征,

提出了基于注意力机制和卷积神经网络的 ACRec (Attention and Convolutional Based Recommendation) 模型, 并利用矩阵分解加强用户和序列的语义关系。本文的主要贡献如下:

- (1) 使用注意力机制和卷积神经网络对序列从动态特征和序列的静态特征两方面进行特征提取;
- (2) 利用矩阵分解增强用户和序列的语义关系;
- (3) 本文在 2 个公共数据集上进行比较实验, 验证 ACRec 的有效性。

1 相关工作

在现实世界中, 用户的行为是连续发生的, 用户下一个点击往往与用户上一个点击的物品相关。近年来, 基于序列推荐的研究也日渐增多。最早对序列进行建模的是基于马尔科夫链的推荐系统, FPMC^[4]是由 Rendle 等人提出的利用马尔科夫链和

作者简介: 陈 昀(1995-), 女, 硕士研究生, 主要研究方向: 推荐系统; 张瑞林(1961-), 男, 博士, 教授, 主要研究方向: 人工智能。

通讯作者: 张瑞林 Email: hzzhangrl@163.com

收稿日期: 2021-05-05

矩阵分解来求解下一项推荐物品。Cheng 等人^[5]在 FPMC 的基础上改进了转移矩阵提出了 FPMC-LR 模型,将转矩阵分解成 2 个小矩阵的乘积。He 等人^[6]通过结合相似度和马尔科夫链提出了 Fossil 推荐模型,利用相似度来计算用户的长期喜好,利用高阶马尔科夫链来计算用户的短期需求。

除了马尔科夫链,许多学者使用深度学习的方法来研究基于序列的推荐。考虑到 RNN 对序列计算的优势,因此许多学者提出了基于 RNN 的推荐系统。Hidasi 等人利用 GRU 对短期会话进行建模,提出了 GRU4Rec 模型。Donkers 等人^[7]设计了融入用户信息的循环神经网络单元,使用户信息在计算过程中可以动态表示。Li 等人^[8]提出了 NARM 模型, NARM 分为编码和解码两个部分,在编码部分对 GRU 计算出的隐藏状态进行全局和局部编码,在解码部分计算物品的点击概率。为了提取序列信息,也有许多学者利用卷积神经网络进行研究。Liu 等人^[9]提出了 CCPM 模型将卷积神经网络应用到推荐算法中,将序列排列成二维矩阵进行特征提取。Liu 等人^[10]提出了 Caser 模型,对卷积核大小进行限制,从水平和垂直两个方向对序列进行了特征提取。Yuan 等人^[11]利用一维卷积对序列进行特征提取。

注意力机制在基于序列的推荐系统中应用得也十分广泛。Zhou 等人提出了 DIN 模型,通过注意力机制计算目标物品与历史序列中物品的相似度,计算了目标物品对历史序列中物品的注意力权值。之后,Zhou 等人^[12]改进了 DIN,提出了 DIEN,利用注意力机制和 GRU 计算用户兴趣的发展变化。Kang 等人^[13]提出了 SASRec 算法模型,利用多层多头自注意力机制来提取历史序列的发展规律。Guo 等人^[14]提出了 IMfOU 算法模型,利用图卷积和注意力机制计算序列的顺序信息,并结合了顺序模型和非顺序模型来更好地计算物品之间的依赖关系。从这些研究可以看出,如何对序列特征进行提取是关键,本文使用注意力机制和卷积神经网络从序列的发展变化特征和序列的静态特征进行特征提取。

2 个性化推荐模型

本节将详细介绍文中提出的个性化推荐模型 (AcRec),如图 1 所示,AcRec 总体由嵌入层、特征提取层、乘积层和全连接层组成。在特征提取层,AcRec 先后使用多头自注意力机制提取序列的动态特征和卷积神经网络来提取序列特征静态特征。

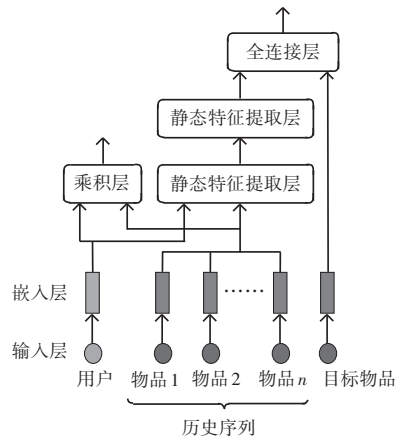


图1 ACRec 的总体结构

Fig. 1 Architecture of ACRec

2.1 嵌入层

由于用户与物品总量大,而用户交互的物品少,因此推荐系统模型的输入是十分稀疏的高维输入。例如用户 $U = \{u_1, u_2, u_3, \dots, u_m\}$, 物品 $I = \{i_1, i_2, i_3, \dots, i_n\}$, 其中 m 和 n 分别为用户和物品的总数。独热码是常用的物品向量表示方式。在独热码中,只有一个位置是 1,其他位置都为 0,通过改变 1 的位置来代表不同的物品。如果用独热码表示每个用户,则 $u_1 = [1, 0, 0, \dots, 0]$, $i_1 = [1, 0, 0, \dots, 0]$ 。如果系统有 1 000 个物品,则一个物品的独热码表式中就会有 999 个 0, 1 个 1,且在这 1 000 个数据中只有一个有用。这样的数据数量大且十分稀疏,不易于计算,因此引入了嵌入层。

嵌入层的思想来源于自然语言处理中用不同的向量来表示每个单词,通过 Skip-gram 或者 CBOV 模型训练单词向量,使得向量能代表单词的意思。在推进系统模型中引入嵌入层思想,用一个向量代表一个物品或用户,使输入数据变得稠密。本文在嵌入层中为每类输入数据都定义了一个嵌入空间,用户的嵌入空间为 $E_U \in R^{U \times k}$, 物品的嵌入空间为 $E_I \in R^{I \times k}$, k 为嵌入维度,且 $k \ll |U|$, $k \ll |I|$, 相比独热码表示,向量表示极大地减少了占用的内存空间。经过嵌入层,第一位用户就表示为 E_U 的第一行数据,第一个物品就表示为 E_I 的第一行数据。

2.2 动态特征提取层

动态特征提取层结构如图 2 所示,研究认为序列中的物品之间是有关联的,如序列中的电脑和键盘、牛奶和面包,为了提取序列中不同物品的关联性,本文使用多头自注意力机制提取序列的关联特征,动态地改变序列中物品的权重。并且由于相同的序列对不同的用户有不同的意义,因此本文使用

注意力机制计算用户对序列中物品的注意力权重。

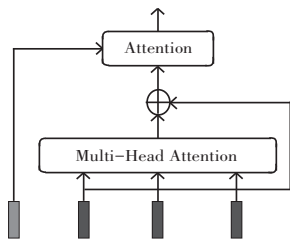


图2 动态特征提取层结构

Fig. 2 Dynamic feature extraction layer

注意力机制是将查询 Q 和键值对 K, V 映射为输出的函数,其计算公式如下所示:

$$Attention = softmax(QK^T)V \quad (1)$$

其中, Q 为当前关注的矩阵、如用户矩阵; V 为需要求解关注度的矩阵、如序列中的物品组成的矩阵; K 是 V 的映射、代表了 V 特性, K 和 V 可以相等, “T” 为转置。计算注意力权重的方法除了公式(1)中使用的点积之外,还可以通过神经网络来计算。

多头自注意力机制在 Transformer 中被提出,通过多个线性变换将输入转换成多组不同的查询、键和值,使得模型能够关注不同方面的注意力权重。Transformer 利用多头自注意力机制来代替 RNN, 计算当翻译不同单词时句子中其他单词的权重。其计算公式如下所示:

$$\begin{aligned} head_i &= Attention(QW_i^Q, KW_i^K, VW_i^V) \\ MultiHead(Q, K, V) &= Concat(head_1, \dots, head_n)W^O \end{aligned} \quad (2)$$

其中, W^Q, W^K, W^V, W^O 为转换矩阵,在自注意力机制中 $Q = K = V$ 都为序列中物品组成的矩阵。多头自注意力机制可以很好地发现序列中物品的关系,因此本文使用多头自注意力机制来动态计算序列中不同物品的权重。经过计算得到序列矩阵 S' , 为了不丢失序列的原始信息,本文将原始的序列矩阵 S 加入到了 S' 中,其计算公式为:

$$S' = S' + S \quad (3)$$

在经过多头注意力机制后,本文计算了用户对序列中物品的注意力权重。研究认为不同用户对相同序列的注重点有所不同,用户自身的特性对用户未来的选择有很大影响。例如,同样是面对序列,电脑、键盘、牛奶、面包,用户 A 会更注意前两项,而用户 B 会更注意后两项。因此即使有相同的历史点击序列,用户 A 和用户 B 未来的点击也是不同的。根据公式(1),本文将用户 A 作为查询,将 B 作为键和值计算注意力权重。

2.3 静态特征提取层

在静态特征提取层,本文通过卷积神经网络提取序列的特征。本文认为不同时间间隔的特征应该分别提取,因此本文用 m 个大小为 $v \times 1$ 的卷积核沿着行与列的方向移动对序列矩阵 S' 进行特征提取, $v \in \{2, \dots, |S|\}$, 通过改变 v 的大小提取不同时间间隔的物品之间的特征,本文用 $F_j^v \in R^{n \times 1}$ 表示第 j 个高度为 v 的卷积核, $1 \leq j \leq |S|$ 。 F_j^v 对历史序列矩阵 S' 从左到右、从上到下地进行卷积计算得到大小为 $(|S| - v + 1) \times k$ 的特征矩阵 C 。然后再将 C 输入池化层,池化层的核大小为 $(K - h + 1) \times 1$, 进行最大池化 $a = \max(C)$, a 表示 C 的最大数值。最后将 $m \times (|S| - 1)$ 个卷积核提取的特征组成 $m \times (|S| - 1) \times k$ 的矩阵,并输入平均池化层,得到大小为 $m \times 1 \times k$ 的特征矩阵 C' , 然后再对 C' 进行重新排列得到 $1 \times (m \times k)$ 的矩阵。

在经过卷积层特征提取后,本文将提取的特征 C' 和用户向量 U 进行非线性变换,融合用户特征和序列特征,计算公式如下所示:

$$Z = ReLU(Concat(C', U)W + b) \quad (4)$$

2.4 乘积层

研究可知,在之前的基于序列的推荐都只对序列特征进行了提取,没有利用用户已经点击过序列中的物品这个信息,使得模型不能明确用户与序列中的物品的关系。为了利用用户与序列中物品的关系,本文提出了乘积层,增强用户和序列中物品的已点击关系。

本文根据矩阵分解算法来计算用户对序列的点击值。在矩阵分解算法中将用户和物品用向量表示,用户对物品的评分为用户向量与物品向量的乘积,通过矩阵分解求解出用户矩阵和物品矩阵就可以预测用户对未评分的物品的评分。因此本文的乘积层利用矩阵分解算法计算用户对历史序列中每个物品的点击概率,并通过 Sigmoid 激活函数使输出限制在 0 和 1 的范围内。由于用户已经点击过历史序列中的物品,所以乘积层的输出都应该接近于 1。乘积层的公式如下所示:

$$y = Sigmoid(US^T) \quad (5)$$

其中, U 和 S 为用户矩阵和序列矩阵。由于乘积层的标签只有一个值 1,因此本文使用交叉熵损失函数计算乘积层的损失值 L_0 。乘积层的损失函数公式为:

$$L_0 = - \sum_{i=0}^n y_i \log y'_i \quad (6)$$

乘积层仅使用在训练阶段调整数据的更新方向,总体的损失函数为 L_0 乘以影响因子 $\tau \in [0,1]$ 加上全连接层的损失值。

3 实验

3.1 实验数据

(1) MovieLens - 1M (<https://grouplens.org/datasets/movielens/1m/>): MovieLens - 1M 是在推荐算法中常用的数据集,包含了1 000 209条用户对电影的评分,3 883部电影和6 040个用户,其中每个用户至少评分过20部电影,电影总共有18个类别,一部电影可以有多个类别。本文把用户看过的电影都标注为1,并随机从用户未看过的电影中选取负样本标注为0,使正负样本的比例为1:1。

(2) Video_Games (http://deepyeti.ucsd.edu/jianmo/amazon/categoryFilesSmall/Video_Games.csv): Video_Games 是 Amazon Review Data 提供的数据集之一。Amazon Review Data 提供了许多数据集,如 Amazon Fashion、Electronics、Books 等。这些数据都包含很多用户对物品的评价和物品信息。Video_Games 是有关游戏产品的数据集,包含了用户对游戏的评价、打分、产品描述等信息。Game 数据集一共包含2 565 349条用户打分、71 982款游戏、1 540 618个用户,平均每个用户有1.7条评论,平均每个物品有35.6条评论。本文对数据集进行了处理,除去数据集中的重复记录和用户交互数量比较少的数据,最终剩余32 232个用户和42 247个游戏,396 983条评论。

本文按用户 ID 和用户评论时间排列数据,把用户评论过的每5个物品作为历史序列,第6个物品作为目标物品,预测用户观看第6个物品概率,并按留一法把每个用户的最后一组序列作为测试集数据。在测试数据中对每个用户通过负采样随机采样了99个不同的负样本,因此每个用户的测试数据一共为100个,其中99个负样本、1个正样本。经过处理的数据集统计详见表1。

表1 数据集统计

Tab. 1 Statistics of datasets

数据集	用户	物品	交互数	训练集	测试集
MovieLens-1M	6 040	3 883	1 000 209	933 769	6 040
Game	32 232	42 247	396 983	204 090	32 232

3.2 对比模型

本文选取了3个推荐系统模型作为本实验的对比模型进行仿真实验,这三个模型是十分具有代表

性的推荐模型。一个是基于矩阵分解型,一个是基于卷积神经网络,一个是基于多头注意力机制,从不同方面对推荐系统进行了研究。

NeuMF^[15]结合了矩阵分解和深度神经网络,通过矩阵分解和深度神经网络分别计算用户对物品的兴趣程度,将传统推荐算法与神经网络相结合,然后连接2部分结果输入全连接层得到最终结果。

Caser利用卷积神经网络提取历史序列特征,通过不同大小的卷积核提取不同维度的特征组合,实现水平特征提取和垂直特征提取,然后将特征和目标物品向量输入全连接层进行计算得到最终结果。

SASRec将Transformer模型转化为推荐算法模型,利用堆叠多层多头自注意力机制和前馈神经网络层交替学习用户历史序列的发展规律,最后将多头自注意力机制的结果与目标物品进行计算预测点击概率。

3.3 评估标准

命中率 $Hit@N$ 和归一化折损累计增益 $NDCG@N$ 是 TopN 推荐问题中常用的评价指标,本文选择这两个评价指标作为实验的评价指标,用于评价算法的好坏。

TopN 推荐为利用模型计算用户对候选物品的点击概率并根据结果进行倒序排列,将前 N 个物品作为推荐列表推荐给用户。 $Hit@N$ 计算 TopN 推荐中是否包含正样本,若是则命中,否则未命中。 $NDCG@N$ 计算在 TopN 推荐中正样本的排序位置,位置越靠前则 $NDCG@N$ 值越大,若不包含正样本则 $NDCG@N$ 值为0。 $Hit@N$ 和 $NDCG@N$ 的计算公式如下所示:

$$Hit@N = \frac{N_{hit}}{N} \quad (7)$$

$$NDCG@N = \frac{\sum_{i=1}^{|U|} \frac{1}{\log_2^{(i+1)}}}{|U|} \quad (8)$$

3.4 参数设置

本文所有实验都选择 Adam 作为优化器,初始学习率都为0.001,正则化参数为 10^{-6} ,批量大小为128,嵌入层维度为64,dropout大小设置为0.3。ACRec的多头自注意力机制的头数设置为4,层数设置为1,垂直卷积核的个数为1,全连接层隐藏层数为3,大小为64、32、16,乘积层的影响因子大小在MovieLens-1M上为0.2,在Video_Games上为0.6。NeuMF的神经网络隐藏层设置为4层,大小分别为64、32、16、8。Caser的每层水平卷积核数位16,垂

直卷积核数为 4。SASRec 的多头自注意力机制和神经网络层交替层数设置为 2,多头自注意力机制的头数也设置为 4。

3.5 分析与结论

实验结果详见表 2,本文分别对比了 Top5 和 Top10 推荐的 $Hit@N$ 和 $NDCG@N$ 值。从表 2 中的数据可以看出不论在 MovieLens-1M 数据集上、还是在 Video_Games 数据集上,本文提出的 ACRec 模型都要优于其他基线对比模型。

表 2 实验结果

Tab. 2 Model performance on public datasets

数据集	评价标准	NeuMF	Caser	SASRec	ACRec
MovieLens-1M	$Hit@5$	0.475 2	0.678 3	0.671 3	0.696 0
	$NDCG@5$	0.319 2	0.509 4	0.509 9	0.525 1
	$Hit@10$	0.663 7	0.804 9	0.795 8	0.813 2
	$NDCG@10$	0.381 7	0.550 6	0.550 7	0.565 0
Video_Games	$Hit@5$	0.356 5	0.355 4	0.317 0	0.459 9
	$NDCG@5$	0.242 7	0.242 5	0.215 6	0.306 6
	$Hit@10$	0.500 3	0.489 2	0.448 8	0.592 4
	$NDCG@10$	0.289 1	0.285 7	0.258 0	0.349 5

在 MovieLens-1M 数据集上,本文提出的 ACRec 明显优于 Caser 和 SASRec 模型,说明从动态和静态两方面提取序列特征的有效性。而 NeuMF 的结果明显弱于其他 3 个模型,这是因为 NeuMF 不是基于序列的,只利用矩阵分解计算了用户和物品的内在特征,而序列给模型提供了更多信息。但在 Video_Games 数据集上 NeuMF 模型优于 Caser 和 SASRec,这是因为 Video_Games 中用户兴趣单一,用户兴趣比序列对用户的选择影响更大。相比 Caser 和 SASRec,本文提出的 ACRec 模型在计算过程中不断融入用户信息,并且加入了乘积层计算用户与序列的关系,因此得到了好的结果,且与 NeuMF 相比有很大提升。

4 结束语

本文主要研究了基于序列的推荐系统。首先利用注意力机制对序列进行动态特征提取,使序列中的物品有不同的权重。然后使用卷积神经网络对序列进行静态特征提取,接着通过改变卷积核的大小提取不同序列长度的特征。并且在模型中增加了乘积层用于加强用户与序列的语义关系。本文在两大公共数据集上进行了对比实验,实验结果证明了本文提出的 ACRec 模型的有效性,无论是 $Hit@N$ 还是 $NDCG@N$ 都要优于其他 3 个对比模型。序列数

据是刻画用户特征的重点,在未来的工作中可以继续对序列提取方面进行研究,并引入知识图谱增加物品信息和关联,来提高推荐的准确率。

参考文献

- [1] 黄剑波,陈方灵,丁友东,等. 基于情感分析的个性化电影推荐[J]. 计算机技术与发展,2020,30(9): 132-136.
- [2] ZHOU G R, ZHU X Q, SONG C R, Ying Fan, et al. Deep interest network for click-through rate prediction [C]// Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York: ACM, 2018: 1059-1068.
- [3] HIDASI B, KARATZOGLOU A, BALTRUNAS L, et al. Session-based recommendations with recurrent neural networks[J]. arXiv Preprint arXiv:1511.06939, 2015.
- [4] 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. New York: ACM, 2010: 811-820.
- [5] CHENG C, YANG H Q, LYU R M, et al. Where you like to go next: Successive point-of-interest recommendation [C]// Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence. California: AAAI Press, 2013: 2605-2611.
- [6] HE R, MCAULEY J. Fusing similarity models with markov chains for sparse sequential recommendation [C]//2016 IEEE 16th International Conference on Data Mining (ICDM). Barcelona: IEEE, 2016: 191-200.
- [7] DONKERS T, LOEPP B, ZIEGLER J. Sequential user-based recurrent neural network recommendations [C]// Proceedings of the Eleventh ACM Conference on Recommender Systems. New York: ACM, 2017: 152-160.
- [8] LI J, REN P J, CHEN Z M, et al. Neural attentive session-based recommendation [C]// Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. New York: ACM, 2017: 1419-1428.
- [9] LIU Q, YU F, WU S, et al. A convolutional click prediction model [C]// Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. New York: ACM, 2015: 1743-1746.
- [10] LIU B, TANG R M, CHEN Y Z, et al. Feature generation by convolutional neural network for click-through rate prediction [C]//The World Wide Web Conference (WWW '19). New York: ACM, 2019: 1119-1129.
- [11] YUAN F J, KARATZOGLOU A, ARAPAKIS I, et al. A simple convolutional generative network for next item recommendation [C]// Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. New York: ACM, 2019: 582-590.
- [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. California: AAAI Press, 2019: 5941-5948.
- [13] KANG W, MCAULEY J. Self-attentive sequential recommendation [C]// 2018 IEEE International Conference on Data Mining (ICDM). Singapore: IEEE, 2018: 197-206.