



细粒度建模用户兴趣的序列化推荐方法

张麒, 吴宾, 孙中川, 叶阳东*

郑州大学计算机与人工智能学院, 郑州 450001

* 通信作者. E-mail: yeyd@zzu.edu.cn

收稿日期: 2021-01-24; 修回日期: 2021-04-16; 接受日期: 2021-09-18; 网络出版日期: 2022-10-10

国家自然科学基金 (批准号: 62176239, 61772475) 和国家青年科学基金 (批准号: 6210071281) 资助项目

摘要 序列化推荐因其实用性和较高推荐精度在近期受到了人们广泛关注. 不同于传统推荐方法, 序列化推荐的核心在于如何基于用户近期交互行为来捕获用户的短期兴趣. 现有工作或者依次考虑用户交互序列中物品之间的成对关系, 忽略了更为重要的多对一关系; 或者将用户最近交互的多个物品视为集合, 忽略了物品之间的相对次序. 这不仅未能充分挖掘物品之间的复杂关系, 而且未准确刻画用户兴趣的演变过程. 为此, 本文提出门耦合胶囊网络 (gating coupled capsule network, GCC), 一种从个体层、联合层以及局部有序性 3 个层面细粒度分析物品间关系对于用户短期兴趣影响的方法. 借助胶囊网络的空间感知能力, GCC 引入个性化胶囊模块建模用户高阶时序信息, 其不仅能够捕获物品间的联合层关系, 也保留了物品间的局部有序性. 另外, 本文在 GCC 中设计了用于建模物品间成对关系的个性化门单元模块, 以此捕获个体层关系对于用户短期兴趣的影响. 在 4 个真实推荐场景下的实验结果表明, GCC 相比于主流序列化推荐方法在多个评价指标上具有显著的性能优势.

关键词 胶囊网络, 序列化推荐, 门单元机制, 隐式反馈, 推荐系统

1 引言

在互联网日益成熟的时代, 推荐系统已经成为在线服务不可或缺的组成部分^[1]. 通过收集挖掘海量的用户行为信息 (例如浏览、点赞和评分等), 推荐系统可以发掘用户潜在需求并提供个性化服务^[2]. 实际生活中用户的交互行为具备天然时序性, 交互对象的变化也实时反映着用户兴趣的演变. 因此, 根据用户近期的行为信息刻画用户短期兴趣对于提升推荐精度和改善用户体验至关重要.

早期的序列化推荐更多关注最近一次交互与下一决策行为之间的关联度^[3], 但用户的即兴决策行为易引发“随机性”问题, 即最近的一次交互并不能准确刻画用户当前短期兴趣. 因此, 合理建模高阶时序信息蕴藏的物品间复杂关系显得尤为重要^[4~13]. 当前的序列化推荐依然面临如下 3 个挑战:

引用格式: 张麒, 吴宾, 孙中川, 等. 细粒度建模用户兴趣的序列化推荐方法. 中国科学: 信息科学, 2022, 52: 1775–1791, doi: 10.1360/SSI-2021-0026
Zhang Q, Wu B, Sun Z C, et al. Fine-grained modeling of user interests for sequential recommendation (in Chinese). Sci Sin Inform, 2022, 52: 1775–1791, doi: 10.1360/SSI-2021-0026

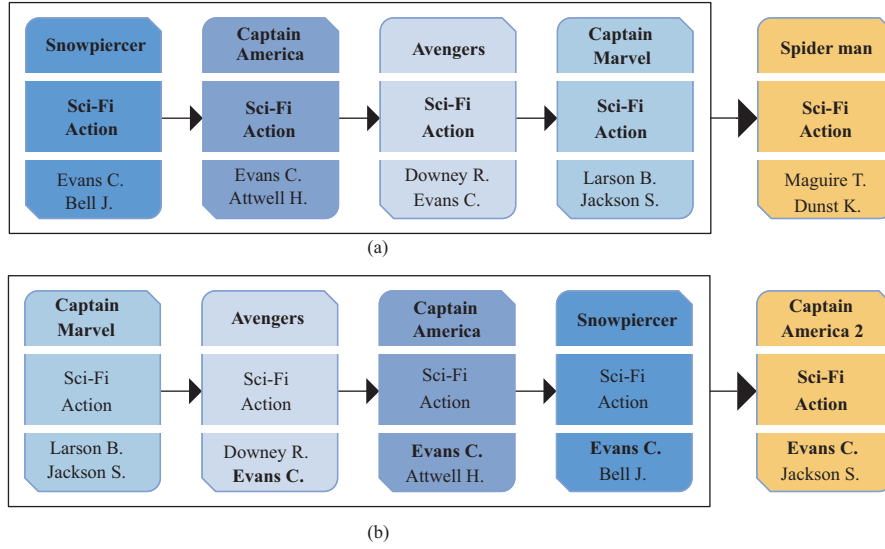


图 1 (网络版彩图) 用户高阶时序信息的局部有序性关系图示 ($L = 4$). 其中蓝色背景系列指代用户电影交互序列, 黄色背景系列表示用户下一时刻交互电影, 每个电影依次包含: 名称、题材和演员信息

Figure 1 (Color online) An illustrative example of the local-order ($L = 4$). The blue series indicate the user action sequence, and the yellow series denote the next action. Each movie contains the name, theme, and actor information. (a) User interest, no protagonist; (b) user interest, focus on Chris Evans

(1) 用户近期多次交互行为中的重点交互对象通常预示着当前的真实兴趣. 例如用户依次购买手机和屏幕保护膜, 则手机的重要性决定了之后更倾向于手机壳或指环等商品. 这种高阶时序信息内重点交互对象同下次决策之间的关联性叫作一对一关系 (也称个体层关系). (2) 用户最近的多交互行为整体与下次决策之间存在关联性. 例如整体考虑序列: 西服、西裤和衬衣, 出于搭配一致性原则, 三者共同决定了下个时刻推荐商务皮鞋会是更合适的选择. 因此准确建模多对一关系 (也称联合层关系) 是序列化推荐的重要任务. (3) 现实生活中用户近期交互对象的不同交互次序往往刻画了不同短期兴趣的演变过程. 例如, 整体考虑图 1(a) 的 4 次交互, 不难发现该用户更偏好超级英雄类科幻电影, 但在不同交互次序的图 1(b) 中, 随着男演员埃文斯 (Evans C.) 戏份增大, 此时用户兴趣解读为偏好埃文斯主演的电影则更为合适, 本文称这种相对次序为局部有序性关系.

尽管近年来研究者已经提出很多建模高阶时序信息的推荐方法, 但是均未同时针对上述 3 个问题设计有效解决方案. 大部分工作或者只注重最基本的个体层关系, 忽略了更重要的联合层关系^[9~13]; 或者将用户高阶时序信息分割到的子序列视为集合^[4~8], 这种“简化”的做法忽略了交互序列的局部有序性特征, 无法准确刻画用户兴趣的演变过程. 值得说明的是, 基于循环神经网络 (recurrent neural network, RNN) 和自注意力机制的方法考虑了局部有序性关系^[9~13]: 前者使用 RNN 结构拟合用户高阶时序信息, 虽一定程度上能够动态刻画用户兴趣, 但难以应对用户兴趣的长期依赖问题. 并且依次考虑最近交互行为的建模方式无法有效应对用户实际交互的“随机性”. 虽然自注意力机制^[13]有效摆脱了对上一次交互的过度依赖并增设位置向量感知局部有序性, 但仅用最近一次交互表示用户兴趣的做法弱化了序列中其他重点交互对象的成对关系, 也限制了联合层关系的提取.

本文针对已有工作的不足, 提出了门耦合胶囊网络 (gating coupled capsule network, GCC), 一种细粒度建模高阶时序信息进而刻画用户兴趣的序列化推荐方法. GCC 首先引入胶囊网络设计个性化胶囊模块, 用于显式捕获具有用户个性化特征的联合层关系. 与此同时, 借助胶囊网络出色的空间感知

能力,在有效建模联合层关系的基础上进一步保留局部有序性,实现对用户短期兴趣的动态刻画.最后,针对高阶时序信息的个体层关系设计个性化门单元模块,该模块能够在建模个体层关系的同时进一步挖掘用户个性化需求,从而更准确地把握用户真实兴趣.为了验证 GCC 应对序列化推荐问题的有效性,在 4 个不同推荐场景的真实数据集上进行实验.大量实验结果表明,相比于当前最流行的序列化推荐方法, GCC 在多个评价指标上具有显著的性能优势.本文主要贡献总结如下.

(1) 提出一种新颖的序列化推荐方法:门耦合胶囊网络 GCC,其通过细粒度分析用户高阶时序信息蕴藏的物品间复杂关系对于用户短期兴趣的影响,实现对用户短期兴趣演变的动态刻画.

(2) 将胶囊网络应用于序列化推荐领域,通过设计个性化胶囊模块建模联合层关系,从而捕获用户兴趣的整体性,并借助胶囊网络的空间感知能力进一步捕捉交互序列的局部有序性;同时针对重点交互对象对用户短期兴趣的影响,设计个性化门单元模块,从而更准确地把握用户真实兴趣.

(3) 在 4 个真实推荐场景的数据集上进行对比实验, GCC 在面对当前流行的序列化推荐方法时均取得更好表现,验证了 GCC 采用细粒度分析高阶时序信息方式刻画用户兴趣的有效性,也体现了胶囊网络解决序列化推荐问题的可行性.

2 相关工作

传统的推荐方法通过学习用户的隐式反馈信息(例如点击、购买和点赞等)更多关注用户的长期兴趣.其中大多数方法利用矩阵分解^[14]计算用户和物品特征向量的内积,以此来建模用户和物品之间的交互关系.例如 BPRMF^[15],是使用矩阵分解算法捕获用户长期兴趣的典型方法.最近, Wang 等^[16]提出一种基于传播用户和物品特征向量实现协同过滤的方法,在对用户-物品图结构的高阶连通性建模的同时,将协同信息显式地注入到特征向量中并取得了显著成绩. He 等^[17]提出 LightGCN 方法对图卷积网络进一步作出改进,通过只保留利于协同过滤的邻域聚合组件,使图卷积网络能够在推荐领域取得更好结果的基础上更易于实现和训练.虽然 NGCF 和 LightGCN 在传统推荐领域取得了优异成绩,但实际表现相对于融合高阶时序信息的序列化推荐方法依然存在差距.

序列化推荐方法的核心在于如何基于用户近期的交互行为来捕获用户的短期兴趣.其中基于马尔科夫(Markov)链的建模方案是经典的序列化推荐方法.例如 FPMC^[3]建模一阶马尔科夫链从而重点获取用户一阶序列特征. Fossil^[4]将马尔科夫链拓展到高阶,并结合物品相似度实现建模高阶时序信息.得益于自然语言处理的发展,基于循环神经网络(RNN)的推荐方法开始进入人们视野,例如 GRU4Rec^[9], GRU4Rec+^[10]等通过依次捕获序列中物品之间的个体层关系提取序列化特征.但是基于 RNN 的方法在每个时刻均严重依赖序列中上一次交互行为,这一做法不仅使其训练效率受到限制,而且约束了联合层关系的提取.自注意力机制(SASRec^[13])的提出有效缓解了 RNN 方法对上一时刻交互的过度依赖问题,但为了检索高阶时序信息中的重点物品同样忽略了联合层关系的建模.最近 Caser^[6]引入卷积神经网络(convolutional neural network, CNN)设计水平和垂直两种形式卷积建模物品联合层和个体层关系. Yu 等^[5]提出 MARank 使用残差网络和注意力机制分别捕获物品联合层和个体层关系.同他们类似, Ma 等^[8]在 HGN 中设计分级门网络捕获物品联合层关系,同时通过计算物品内积实现个体层关系建模.虽然上述方法取得了较好表现,但忽略了序列中最近物品的局部有序性.

“胶囊(capsule)”一词由 Hinton 及其团队在 2011 年首次引入计算机领域^[18],后经与动态路由算法结合形成胶囊网络(capsule networks, CapsNet)^[19],如今已流行于计算机视觉^[20]、模式识别^[21]以及知识图谱^[22]等多个领域.最精简的 CapsNet 是基于卷积操作以完成图片特征提取为目标发展

而来的。但是, 由于每一个胶囊均是由若干个小的神经元构成的, 因此相对于传统 CNN 提取标量特征而言, 胶囊网络的空间感知能力决定了所提取的特征具有感知方向的“等价性”特点。在此基础上, SegCaps^[23] 已经验证采用胶囊网络建模目标空间关系比传统的基于 CNN 的方法更有效。CapsE^[22] 成功将胶囊网络应用于知识图谱来建模“实体-关系-实体”三元组关系。MIND^[24] 将动态路由机制应用于工业届解决用户多兴趣推荐任务。JTCN^[25] 将胶囊网络与注意力机制相结合来缓解推荐系统冷启动问题。虽然胶囊网络已经在多个领域崭露头角, 但尚未用于解决序列化推荐问题。

受上述方法启发, 本文提出一种新的序列化推荐方法。其通过设计个性化胶囊模块用于建模用户高阶时序信息, 进而在更有效地捕获物品联合层关系的同时, 保留序列中物品的局部有序性。同时针对高阶时序信息的个体层关系设计个性化门单元模块, 从而尽可能准确地刻画用户兴趣。

3 门耦合胶囊网络

3.1 相关定义

定义1 (用户-物品交互矩阵) 给定用户集合 $\mathcal{U} = \{u_1, u_2, \dots, u_{|\mathcal{U}|}\}$ 和物品集合 $\mathcal{I} = \{i_1, i_2, \dots, i_{|\mathcal{I}|}\}$, 则根据用户是否同某个物品产生交互, 可定义形状为 $|\mathcal{U}| \times |\mathcal{I}|$ 的用户-物品交互矩阵 $\mathbf{R}$:

$$r_{ui} = \begin{cases} 1, & \text{如果用户 } u \text{ 同物品 } i \text{ 产生交互,} \\ 0, & \text{反之.} \end{cases}$$

定义2 (用户序列化行为) 按时间先后顺序描述用户 u 完整的历史交互记录, 即单个用户的序列化行为信息, 定义为 $\mathcal{S}^u = \{\mathcal{S}_1^u, \mathcal{S}_2^u, \dots, \mathcal{S}_{|\mathcal{S}^u|}^u\}$, 则所有用户的序列化行为定义为集合 $\mathcal{A} = \{\mathcal{S}^u | u \in \mathcal{U}\}$ 。

定义3 (序列化推荐任务) 给定用户 u , 物品集合 $\mathcal{I}$ 和用户交互序列 $\mathcal{S}^u$ 。则序列化推荐任务定义为: 根据用户 u 已交互的物品序列 $\mathcal{S}^u$, 从该用户未曾交互过的物品集 $\mathcal{I} \setminus \mathcal{S}^u$ 中推荐一组按偏好程度排序的目标物品。

3.2 模型结构

(1) **特征向量映射**. 本文将用户集合 $\mathcal{U}$ 和物品集合 $\mathcal{I}$ 中的所有对象映射为相应特征向量构成特征矩阵 $\mathbf{P}$ 和 $\mathbf{Q}$ 。另外, 与已有工作^[5~8]保持一致, 额外初始化一个物品矩阵 $\mathbf{E}$ 用来捕获序列化特征, 如式 (1) 所示:

$$\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_{|\mathcal{U}|}], \quad \mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_{|\mathcal{I}|}], \quad \mathbf{E} = [\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_{|\mathcal{I}|}], \quad (1)$$

其中 $\mathbf{P} \in \mathbb{R}^{K \times |\mathcal{U}|}$, $\mathbf{Q}, \mathbf{E} \in \mathbb{R}^{K \times |\mathcal{I}|}$, K 为特征向量维度。具体如图 2 向量映射层所示。另外, 输入 GCC 的高阶物品矩阵 $\mathbf{S}_{u,l} \in \mathbb{R}^{K \times L}$, 将每个用户 u 的完整交互序列 $\mathcal{S}^u$ 按照滑动窗口策略顺序选取 t 时刻之前 L 个连续物品, 再根据索引选取矩阵 $\mathbf{E}$ 对应向量所得。

(2) **联合层关系建模**. 高阶时序信息蕴藏的物品联合层关系体现了用户最近时段的 L 次交互行为整体与下一时刻决策行为之间的关联性。例如整体考虑用户观看《钢铁侠 1》和《钢铁侠 2》后要比单独考虑观看《钢铁侠 1》或《钢铁侠 2》后更有可能观看下一部钢铁侠电影。已有的工作通过设计残差网络^[5]、分级门网络^[8]或组合卷积^[6]等方式捕获联合层物品关系, 但都将连续的 L 次交互视为了集合。在实际生活中, 同一组交互对象的不同交互次序往往描述了用户不同短期兴趣的演变过程, 因此在保留局部有序性的前提下捕获联合层关系是必要的。最近胶囊网络因其突出的空间感知能力在计

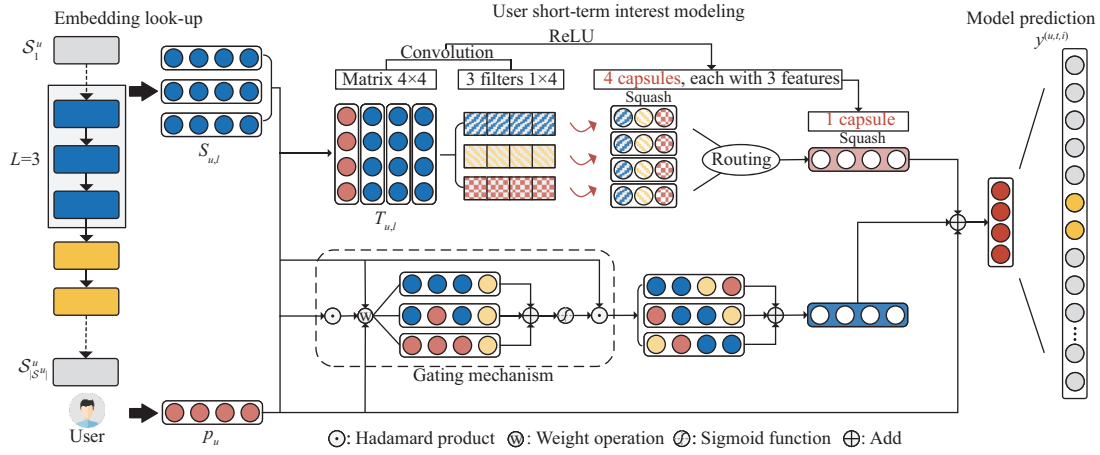


图 2 (网络版彩图) GCC 结构图. GCC 主要包括向量映射、建模用户短期兴趣和模型预测 3 个组件

Figure 2 (Color online) An illustration of the GCC model architecture. GCC consists of three major components: the embedding layer, the user short-term interests modeling, and the model prediction

计算机视觉^[20]以及知识图谱^[22]等领域取得显著进展. 所谓空间感知能力, 指的是胶囊网络在卷积计算后将标量结果向量化, 再通过与权重矩阵线性变换赋予输入向量额外空间信息, 进一步通过动态路由算法计算出蕴含正确方向信息的潜在特征. 本文将胶囊网络这一特性在序列化推荐任务中定义为局部有序性. 换言之, 胶囊网络相对于卷积神经网络能够在有效捕获物品间联合层关系的基础上额外获取物品的相对位置信息. 受此启发, 本文设计一种个性化胶囊模块用于显式捕获物品联合层关系并感知局部有序性特征.

经典的胶囊网络包含“卷积层”和“路由层”两层结构, 本文将每个 L 阶物品特征矩阵 $S_{u,l} \in \mathbb{R}^{K \times L}$ 和用户特征向量 $p_u \in \mathbb{R}^K$ 进行拼接, 构成“个性化物品矩阵” $T_{u,l} \in \mathbb{R}^{K \times (L+1)}$, 输入后续模块以便实现用户个性化. 在卷积层中, 将矩阵 $T_{u,l}$ 的每一行均视为 1 个特征序列 (共 K 个特征序列), 不同的特征序列均蕴含一定的局部有序性特征. 为了便于胶囊网络充分提取物品间的联合层关系并从不同角度编码物品位置信息, 本文设计了 H 个卷积核 $V^{(h)} \in \mathbb{R}^{1 \times (L+1)}$, 分别从不同角度对每一个特征序列均采用水平卷积操作提取潜在特征. 之后, GCC 将同一特征序列对应的卷积结果构成新的胶囊向量, 以此实现每个胶囊均编码所有物品位置信息的目的. 具体计算过程如式 (2) 所示:

$$F_{\text{Conv}} = [f^{(1)}, \dots, f^{(h)}, \dots, f^{(H)}], \quad f^{(h)} = \text{squash} \left(\text{ReLU}(T_{u,l} * V^{(h)} + b) \right), \quad (2)$$

其中 $F_{\text{Conv}} \in \mathbb{R}^{K \times H}$, $f^{(h)} \in \mathbb{R}^K$, $*$ 代表卷积操作, 偏置项 $b \in \mathbb{R}^K$, $\text{squash}(\cdot)$ 代表非线性激活函数, 在每一层网络输出前将向量归一化 ($\text{squash}(x) = \frac{\|x\|^2}{1 + \|x\|^2} \cdot \frac{x}{\|x\|}$, x 代表输入向量).

由于 CNN 卷积之后设置的最大值池化只关注最突出的信息, 而且均值池化会使信息变得模糊, 这直接造成原始物品序列部分信息的丢失, 均无法准确描述序列的局部有序性关系. 而动态路由机制在运算伊始即通过与权重矩阵线性变换融入物品空间信息得到高层次胶囊, 再将该胶囊与迭代计算的耦合系数作点积判断低层次与高层次胶囊方向是否一致, 从而在路由过程中增强方向一致向量的特征表示, 弱化方向不一致向量的特征表示, 进而随着动态路由算法的推进能够显著增强蕴含正确序列方向信息的胶囊, 弱化蕴含错误序列方向信息的胶囊, 最终准确计算出物品之间的相对次序, 得到序列的局部有序性关系. 因此, GCC 在卷积层之后设计动态路由机制在感知局部有序性的基础上进一步提取深层次抽象特征. 值得注意的是, 为了突出胶囊网络建模用户高阶时序信息的有效性, 同时尽可

能减少胶囊网络输出的参数量, 本文控制每个矩阵 $\mathbf{T}_{u,l}$ 在经过胶囊计算后只提取出 1 个胶囊 (向量). 具体而言, 1 次路由过程中首先将卷积层输出的所有胶囊 $\mathbf{c}_u^{(k)} \in \mathbb{R}^H, k \in 1, 2, \dots, K$ 分别与对应权重 $\mathbf{W}_c \in \mathbb{R}^{K \times H}$ 进行线性变换得到向量 $\hat{\mathbf{c}}_u^{(k)} \in \mathbb{R}^K$. 再将其与迭代更新的耦合系数相乘改变当前胶囊重要程度, 之后将向量 $\hat{\mathbf{c}}_u^{(k)}, k \in 1, 2, \dots, K$ 求和得到胶囊向量 $\mathbf{s}_u$. 最后对该胶囊进行非线性激活函数处理得到输出向量 $\mathbf{f}_{\text{Caps}}$. 主要计算过程为

$$\mathbf{f}_{\text{Caps}} = \text{squash}(\mathbf{s}_u), \quad \mathbf{s}_u = \sum_{k=1}^K \beta_k \mathbf{W}_c \mathbf{c}_u^{(k)}, \quad (3)$$

其中 $\mathbf{f}_{\text{Caps}} \in \mathbb{R}^K$ 代表个性化胶囊模块的输出, β_k 代表动态路由机制计算的耦合系数, 表示胶囊 $\mathbf{c}_u^{(k)}$ 的重要程度. 值得说明的是, 在式 (3) 中选择按行整合, 可以使动态路由算法直接作用于不同特征序列对应的潜在特征之间, 由于不同特征序列对应的潜在特征在路由过程中会得到不同程度增强或弱化, 所以此举能够减小蕴含错误方向信息胶囊对最终结果的影响. 算法 1 描述了详细的计算流程.

算法 1 Dynamic routing process

Input: $\mathbf{F}_{\text{Conv}}$.

```

1: for all capsule  $k \in$  the first layer do
2:    $\alpha_k \leftarrow 0$ ;
3: end for
4: for iteration = 1, 2, ...,  $m$  do
5:    $\beta_k \leftarrow \text{softmax}(\alpha_k)$ ;
6:    $\mathbf{s}_u \leftarrow \sum_{k=1}^K \beta_k \hat{\mathbf{c}}_u^{(k)}$ ;
7:    $\mathbf{f}_{\text{Caps}} = \text{squash}(\mathbf{s}_u)$ ;
8:   for all capsule  $k \in$  the first layer do
9:      $\alpha_k \leftarrow \alpha_k + \hat{\mathbf{c}}_u^{(k)} \cdot \mathbf{f}_{\text{Caps}}$ ;
10:  end for
11: end for

```

Output: $\mathbf{f}_{\text{Caps}}$.

(3) 个体层关系建模. 由于用户最近多次交互行为中的重点交互对象预示着用户当前的真实兴趣, 同时用户交互行为的“随机性”现象和稀疏性问题造成了用户最近多次连续交互并不一定均同下一次决策保持着密切关联, 因此捕捉高阶时序信息中重点交互对象同下一决策之间的一对一关联性有助于准确刻画用户当前的真实兴趣 (体现为个体层关系). 例如用户购买手机之后, 大概率会进一步选购手机壳或屏幕保护膜类商品. 对于这种成对关系, 已有工作要么忽略了对显式建模 (如 CosRec^[7]), 要么对用户的个性化特征考虑欠佳 (如 MARank^[5], HGN^[8]). 受线性门单元 (gated linear unit, GLU)^[26] 启发, GCC 设计个性化门单元模块用于显式捕获个体层关系. 原始 GLU 计算过程为

$$\text{GLU}(\mathbf{X}) = (\mathbf{X} * \mathbf{W} + \mathbf{b}) \odot \sigma(\mathbf{X} * \mathbf{V} + \mathbf{c}), \quad (4)$$

其中 $\mathbf{X}$ 代表输入矩阵, $\mathbf{W}, \mathbf{V}, \mathbf{b}, \mathbf{c}$ 代表可学习的参数, σ 代表 sigmoid 函数 ($\sigma(x) = 1/(1 + e^{-x})$), $*$ 代表卷积操作, $\odot$ 代表哈达玛积 (Hadamard product) 计算. 不难发现基于卷积的 GLU 不仅计算消耗大, 而且不适用于以单个物品为操作对象的个体层关系建模. 另外, 现实生活中的部分人气用户和热门物品具有一定的普适性, 但并非所有的用户都热衷于热门物品. 因此, GCC 设计个性化门单元模块显式捕获具有用户个性化信息的个体层关系. 具体而言, 该模块通过为物品项、用户项以及用户物品交互项分别分配独立权重参数, 使不同因子项能够直接被门单元机制处理. 在这一过程中, 通过用户和物

品的实际交互数据能够实时更新权重,从而增强重点物品的特征表示能力以及用户与当前物品的个性化关系,进而在缓解用户被热门物品干扰的基础上有效捕获物品间的一对一关系. 式 (5) 描述了个性化门单元模块的详细计算过程:

$$\mathbf{g}^{\text{Gating}} = \sum_{l=1}^{|L|} \mathbf{e}_l \odot \sigma \left(\mathbf{W}_g^{(1)} \cdot \mathbf{e}_l + \mathbf{W}_g^{(2)} \cdot \mathbf{p}_u + \mathbf{W}_g^{(3)} \cdot (\mathbf{p}_u \odot \mathbf{e}_l) \right), \quad (5)$$

其中 $\cdot$ 代表矩阵乘法, $\mathbf{W}_g^{(1)}, \mathbf{W}_g^{(2)}, \mathbf{W}_g^{(3)} \in \mathbb{R}^{K \times K}$ 代表可学习的权重参数, $\mathbf{p}_u \in \mathbb{R}^K$ 代表用户 u 的特征向量, $\mathbf{e}_l \in \mathbb{R}^K$ 代表已交互物品的特征向量.

(4) **用户长短期兴趣融合.** 本文研究重点在于动态刻画用户短期兴趣, 因此对于用户长期兴趣的捕获同主流序列化推荐方法^[3,5,8]保持一致, 即采用经典的矩阵分解算法. 在对用户不同类型兴趣捕获完毕后, GCC 将多类型兴趣特征以“求和”的形式融合得到用户 u 在目标时刻 t 购买物品 i 的预测评分值 $y^{(u,t,i)}$, 详细计算过程如下:

$$y^{(u,t,i)} = \underbrace{\mathbf{p}_u^T \cdot \mathbf{q}_i}_{\text{长期兴趣}} + \underbrace{\mathbf{f}_{\text{Caps}}^T \cdot \mathbf{q}_i + \mathbf{g}_{\text{Gating}}^T \cdot \mathbf{q}_i}_{\text{短期兴趣}}, \quad (6)$$

其中第 1 部分用于建模用户长期兴趣, 第 2 部分为本文设计的个性化胶囊模块和门单元模块, 用于捕获用户的短期兴趣.

3.3 训练优化

给定用户 u 和最近的连续 L 次交互行为 $\{\mathcal{S}_{t-L}^u, \dots, \mathcal{S}_{t-1}^u\}$, 则该用户在目标时刻 t 与物品 i 产生交互的概率为

$$\text{Prob}(\mathcal{S}_t^u | \mathcal{S}_{t-1}^u, \mathcal{S}_{t-2}^u, \dots, \mathcal{S}_{t-L}^u) = \sigma(y^{(u,t,i)}), \quad (7)$$

其中 $\sigma(x) = 1/(1 + e^{-x})$ 为 sigmoid 函数. 假设所有用户相互独立, 以 Θ 表示 GCC 所有可学习的模型参数, 则通过对式 (8) 求最大似然值可以学习 GCC 相关模型参数:

$$\arg \max_{\Theta} \prod_{u \in \mathcal{U}} \prod_{t=L+1}^{|\mathcal{S}^u|} \sigma(y^{(u,t,i)}) \prod_{j \neq i} (1 - \sigma(y^{(u,t,j)})) \text{Prob}(\Theta). \quad (8)$$

对式 (8) 取负对数可得式 (9), 即 GCC 所对应目标函数, 该函数亦称为二分类交叉熵损失 (binary cross-entropy):

$$\mathcal{L} = - \sum_{u \in \mathcal{U}} \sum_{t=L+1}^{|\mathcal{S}^u|} \log(\sigma(y^{(u,t,i)})) + \sum_{j \in \mathcal{T} \setminus \mathcal{S}^u} \log(1 - \sigma(y^{(u,t,j)})) + \frac{\lambda}{2} \|\Theta\|^2, \quad (9)$$

其中 $\Theta = \{\mathbf{P}, \mathbf{Q}, \mathbf{E}, \mathbf{V}, \mathbf{W}_c, \mathbf{W}_g, \mathbf{b}\}$ 代表最小化目标函数过程中所学习的模型参数, λ 代表正则化系数. 在训练过程中, GCC 根据文献 [27] 所制定策略迭代优化损失函数, 在每个训练回合均针对批量样本的损失函数 $\mathcal{L}$ 计算梯度, 并在此基础上通过反向传播更新所有模型参数值.

3.4 模型结构分析

(1) **复杂性分析.** GCC 模型参数量主要包括两部分: 与数据集规模呈正相关的参数项和独立参数项, 前者主要来自 GCC 初始化特征矩阵, 后者主要来自个性化胶囊模块和个性化门单元模块. 记数据集含 $|\mathcal{U}|$ 个用户, $|\mathcal{I}|$ 个物品, 特征向量维度为 K , 则 GCC 第 1 部分模型参数数量为 $(|\mathcal{U}| + 2|\mathcal{I}|)K + |\mathcal{I}|$, 第 2 部分模型参数数量为 $(L + 2)H + (3 + H)K^2$, 其中 L 为高阶序列阶数, H 为卷积核数. 此外, 本文中最强基准方法 HGN 的模型参数数量为 $(|\mathcal{U}| + 2|\mathcal{I}| + L + 1)K + |\mathcal{I}| + 2K^2$, 典型的序列化推荐方法 FPMC 为 $(|\mathcal{U}| + 3|\mathcal{I}|)K$. 当代入实际数据集时, GCC 和 HGN 的参数量是相近的, 且远少于 FPMC. 例如 Beauty 数据集, HGN 为 1050 万, FPMC 为 1520 万, 而 GCC 为 1060 万.

GCC 的时间消耗主要来自个性化胶囊模块、个性化门单元模块和矩阵分解算法 3 部分. 对于个性化胶囊模块而言, 由于涉及卷积和动态路由计算, 因此时间复杂度为 $O(KH(L + m) + mK^2)$, 其中 m 为动态路由迭代次数. 对于个性化门单元模块而言, 只涉及矩阵运算, 因此时间复杂度为 $O(LK^2)$. 最后, 将上述两者同矩阵分解算法对应的 $O(K|\mathcal{R}^+|)$ 相加 ($|\mathcal{R}^+|$ 表示用户物品交互矩阵 $\mathbf{R}$ 中的非 0 元素个数), 即可得到 GCC 训练时对应的时间复杂度 $O(K(L + m)(H + K) + K|\mathcal{R}^+|)$.

(2) **与主流方法的关联性分析.** 为了论证主流序列化推荐方法与 GCC 之间存在的泛化关系, 选取结构具有代表性的 FPMC, MARank 和 Caser 同 GCC 进行类比.

(i) 相比 FPMC. 如式 (10) 所示, FPMC 通过矩阵分解捕获用户长期兴趣, 并应用一阶马尔科夫链建模用户短期兴趣, 最后以求和的形式融合用户多类型兴趣特征. 该方法受一阶转移矩阵限制无法应对高阶交互序列, 因此, 将 GCC 去除联合层关系建模并只在最近一次交互范围内建模个体层关系时, GCC 弱化为 FPMC 方法.

$$\hat{y}_{\text{FPMC}}^{(u,t,i)} = \underbrace{\mathbf{p}_u^T \cdot \mathbf{q}_i}_{\text{长期兴趣}} + \underbrace{\mathbf{g}_{S_{t-1}^u}^T \cdot \mathbf{q}_i}_{\text{短期兴趣}}. \quad (10)$$

(ii) 相比 MARank. 如式 (11) 所示, MARank 利用深度神经网络将 FPMC 拓展到高阶, 并引入残差网络和注意力机制分别建模物品联合层和个体层关系. 但无论是残差网络还是注意力机制均无法感知序列中物品的相对位置信息. 因此, MARank 实质上为 GCC 删除局部有序性特征的一个特例.

$$\hat{y}_{\text{MARank}}^{(u,t,i)} = \underbrace{\mathbf{p}_u^T \cdot \mathbf{q}_i}_{\text{长期兴趣}} + \underbrace{\mathbf{f}_L^T \cdot \mathbf{q}_i + \mathbf{g}_{S_{t-1}^u}^T \cdot \mathbf{q}_i}_{\text{短期兴趣}}. \quad (11)$$

(iii) 相比 Caser. 如式 (12) 所示, 将物品特征矩阵视为“图片”采取水平和垂直两种形式卷积提取局部特征捕获用户短期兴趣是 Caser 的主要贡献. 对于不同卷积方式得到的多类型用户兴趣, Caser 采取“拼接”策略进行融合, 这一做法引入了大量模型参数. 另外, CNN 虽可以高效提取局部特征, 但受其自身限制无法应对方向性问题. 因此, Caser 本质上为 GCC 不感知局部有序性关系的弱化版特例.

$$\hat{y}_{\text{Caser}}^{(u,t,i)} = \underbrace{\mathbf{W} \begin{bmatrix} \tilde{\mathbf{o}} \\ \mathbf{p}_u^T \end{bmatrix} + \mathbf{b}}_{\text{长期兴趣}}, \quad \tilde{\mathbf{o}} = \text{ReLU} \left(\underbrace{\mathbf{W}' \begin{bmatrix} \mathbf{o}_v \\ \mathbf{o}_h \end{bmatrix} + \mathbf{b}'}_{\text{短期兴趣}} \right). \quad (12)$$

表 1 数据集的统计信息
Table 1 Statistics of the evaluation datasets

Dataset	#User	#Item	#Action	Sparsity (%)
Cellphone	9534	53479	139141	99.97
Beauty	12595	75253	222897	99.98
Gowalla	13149	14009	535650	99.71
ML-1M	6040	3706	1000209	95.53

4 实验设置与结果分析

4.1 数据集介绍

为了验证 GCC 解决序列化推荐问题的有效性, 本文在不同推荐场景 (商品、兴趣点、电影) 的公开数据集上进行大量实验. 其中, 商品推荐场景的亚马逊数据集^[28] 包含了 1996 年 5 月 ~ 2014 年 7 月用户对不同商品的购买记录. 本文采用了其中两个代表性数据集 Cellphone 和 Beauty. 兴趣点推荐场景的 Gowalla 数据集^[29] 记录了社交网站 gowalla.com 中已注册用户的兴趣点签到信息. 电影推荐场景的 ML-1M 数据集^[30] 囊括了 MovieLens 电影网站收集的用户观影记录, 包含了大约 100 万条交互数据.

在数据集预处理方面, 只保留了其中 4 个特征: 用户编号、物品编号、评分数值、交互发生时间. 评分项采取隐式反馈设置, 将用户对物品评分大于等于 1 分的数据均视为正样本, 将用户对物品评分为 0 或缺失评分的数据视为负样本. 最后, 在保留所有物品以保持数据长尾性的基础上过滤掉交互次数小于 10 次的用户. 表 1 展示了数据集详细的统计信息.

4.2 评价指标

在推荐任务中, Precision, Recall, MAP 和 NDCG 是序列化推荐常用的评价指标^[31~33], 其分别从不同角度说明了方法实际性能. 因此本文选取上述 4 个指标来尽可能全面地评价方法性能. 另外, 上述指标均为数值越大表示性能越强. 相关计算如式 (13) 所示:

$$\begin{aligned}
 \text{Precision@}N &= \frac{1}{|\mathcal{U}|} \sum_{u=1}^{|\mathcal{U}|} \frac{|l_{\text{rec}}^u \cap l_{\text{tes}}^u|}{N}, & \text{MAP@}N &= \frac{1}{|\mathcal{U}|} \sum_{u=1}^{|\mathcal{U}|} \frac{\sum_{a=1}^N P_u(a) \times \delta_u(a)}{\min\{N, |l_{\text{tes}}^u|\}}, \\
 \text{Recall@}N &= \frac{1}{|\mathcal{U}|} \sum_{u=1}^{|\mathcal{U}|} \frac{|l_{\text{rec}}^u \cap l_{\text{tes}}^u|}{|l_{\text{tes}}^u|}, & \text{NDCG@}N &= \frac{1}{|\mathcal{U}|} \sum_{u=1}^{|\mathcal{U}|} \frac{\sum_{a=1}^N \frac{2^{\delta_u(a)} - 1}{\log_2(a+1)}}{1/\log_2(a+1)},
 \end{aligned} \tag{13}$$

其中 l_{rec}^u 表示系统向用户 u 推荐的个性化列表, l_{tes}^u 表示用户 u 在测试集中的实际交互列表, $P_u(a)$ 表示用户 u 对应的个性化推荐列表中前 a 个位置的准确率, $\delta_u(a)$ 表示二分类指示函数, 当用户 u 购买了推荐列表中第 a 个物品时为 1, 否则为 0.

4.3 基准方法的选取及实验设置

本文选取的基准方法包括 BPRMF, FPMC, Caser, LightGCN, SASRec, MARank 和 HGN. 此外, 关于上述方法建模高阶时序信息的能力如表 2 所示.

BPRMF^[15]. 使用矩阵分解计算用户 - 物品交互信息, 以此捕获用户长期兴趣的传统推荐方法.

FPMC^[3]. 在最近一次交互范围内捕获个体层关系的序列化推荐方法.

表 2 各基准方法建模高阶时序信息能力对比
Table 2 The characteristics of baselines and GCC

Characteristic	BPRMF	FPMC	Caser	LightGCN	SASRec	MARank	HGN	GCC
Union-level	✗	✗	✓	✗	✗	✓	✓	✓
Individual-level	✓	✓	✓	✓	✓	✓	✓	✓
Local-order	✗	✗	✗	✗	✓	✗	✗	✓

Caser [6]. 基于 CNN 设计水平和垂直两种形式卷积捕获联合层和个体层关系的序列化推荐方法.

LightGCN. He 等 [17] 最近提出的传统推荐方法, 通过只保留邻域聚合组件实现更高效地捕获用户长期依赖.

SASRec [13]. 基于自注意力机制的序列化推荐方法, 运用自注意力机制建模物品的个体层关系.

MARank [5]. 使用残差网络和注意力机制分别捕获联合层和个体层关系的序列化推荐方法.

HGN [8]. 使用分级门网络捕获联合层关系, 并设计内积模块捕获个体层关系的序列化推荐方法.

GCC. 本文提出的序列化推荐方法, 通过设计个性化胶囊模块建模物品的联合层关系并保留最近物品的局部有序性特征, 同时设计个性化门单元模块捕获物品的个体层关系.

除此之外, 参与本文对比实验的传统推荐和序列化推荐方法还包括: NGCF [16], Fossil [4], Cos-Rec [7] 和 GRU4Rec [9], 但由于其表现不如上述基准方法, 受篇幅限制不再展示其详细实验结果.

在实验中, 所有用户交互序列按时间先后截取前 70% 作为训练集, 接下来 10% 用作验证集, 剩下的 20% 作为测试集. 为了尽可能使对比实验公平合理, 本文将上述 7 个基准方法以及 GCC 均用 Tensorflow 1.14.0 进行复现. 所有推荐方法的特征向量维度 K 若无特殊说明均设置为 64, 批处理数据量设置为 512. 所有超参数均采用网格搜索法选出最优参数. 从 $\{0, 1, 10^{-1}, \dots, 10^{-5}\}$ 寻找最佳学习速率和正则化系数. 从 $\{1, 2, \dots, 7\}$ 寻找最佳高阶时序信息阶数 L . 从 $\{4, 8, 16, 32, 64\}$ 和 $\{1, 2, 4, 8, 16\}$ 分别寻找 Caser 最佳水平和垂直卷积核数. 从 $\{1, 2, 3\}$ 和 $\{50, 100, 150, 200\}$ 分别寻找 SASRec 最佳堆叠次数和最大序列长度. 从 $\{0, 2, 4, 8\}$ 和 $\{1.0, 2.0, \dots, 5.0\}$ 分别寻找 MARank 最佳网络层数和梯度截断临界值. 从 $\{1, 2, 3\}$ 分别寻找 Caser 和 HGN 最佳目标物品个数. 对于 GCC 而言采取上述相同的调参范围. 另外, 从 $\{4, 8, 16, 32, 64\}$ 寻找最佳卷积核数, 从 $\{1, 2, 3, 4, 5\}$ 寻找最佳路由迭代次数. 所有实验均在配置 Intel Xeon(R) E5-2640 处理器、Geforce TITAN V 显卡、256GB 运行内存和 Linux 4.4.0 操作系统的服务器上完成.

4.4 实验结果分析

实验 1 与主流推荐方法对比实验分析

GCC 同其他主流推荐方法的对比实验结果如表 3 和图 3 所示, 包括了 $K = 16$ 和 64 两组不同特征向量维度以及 $N = \{5, 10, 15, 20\}$ 4 组不同推荐列表长度. 通过分析实验结果可得如下结论.

- (1) 所有方法在不同推荐列表长度下均呈现相同趋势: Precision@ N 随着 N 增大而递减; NDC-G@ N 随着 N 增大而递增. 并且在不同推荐列表长度下 GCC 均取得了最佳表现, 验证了门耦合胶囊网络建模用户兴趣的有效性.
- (2) 所有方法在前两个数据集的实验结果数值上要明显小于后两个数据集的结果. 主要原因归咎于前两个数据集稀疏性问题较为严重, 因此所呈现的数据缺少足够多的用户连续交互记录, 导致了能够学习到的有效信息相对较少. 此外, 表 3 中对于 Cellphone 和 Beauty 的实验结果数值上均乘以 100%.

表 3 不同推荐方法在 4 个数据集上当推荐列表长度为 10 时的性能表现

Table 3 Performance of different methods on four datasets with $N = 10$

Dataset	Metric@10	BPRMF	FPMC	Caser	LightGCN	SASRec	MARank	HGN	GCC	Improve (%)
Cellphone ($K = 16$) (%)	Precision	0.4352	0.5695	0.5045	0.5768	0.4950	0.6010	0.6251*	0.7394	18.29
	Recall	1.6765	1.9746	1.8105	1.9871	1.8723	2.1247	2.1885*	2.6880	22.82
	MAP	0.5857	0.8591	0.7029	0.8832	0.6811	0.8069	0.9072*	1.0605	16.90
	NDCG	1.0741	1.4370	1.2298	1.5185	1.2387	1.4364	1.5574*	1.8364	17.91
Cellphone ($K = 64$) (%)	Precision	0.5391	0.6503	0.5800	0.7331	0.7153	0.8737	0.8821*	0.9534	8.09
	Recall	2.0249	2.3063	2.1023	2.8595	2.7118	3.2327	3.2347*	3.5170	8.73
	MAP	0.8415	1.0194	0.8438	1.1823	1.0365	1.2855	1.3170*	1.4873	12.93
	NDCG	1.3991	1.6821	1.4527	1.9662	1.8191	2.2058	2.2612*	2.4887	10.06
Beauty ($K = 16$) (%)	Precision	0.5295	0.7240	0.6748	0.6693	0.5367	0.7375	0.7645*	0.8511	11.32
	Recall	1.5872	1.8183	1.7128	1.7866	1.6374	1.9623*	1.8679	2.2524	14.78
	MAP	0.5743	0.7837	0.7006	0.7119	0.5850	0.8041	0.8627*	0.9027	4.63
	NDCG	1.0748	1.4262	1.2941	1.3606	1.1355	1.4674	1.4870*	1.5951	7.27
Beauty ($K = 64$) (%)	Precision	0.6899	0.9495	0.8122	0.7764	0.8741	0.9829	1.0416*	1.1822	13.49
	Recall	2.2319	2.5720	1.9813	2.4799	2.3466	2.8102*	2.7691	3.1917	13.57
	MAP	0.8087	1.0476	0.7885	0.9342	0.9211	1.0556	1.1982*	1.3241	10.50
	NDCG	1.5085	1.9015	1.4476	1.7103	1.7359	1.9985	2.0921*	2.3689	13.23
Gowalla ($K = 16$)	Precision	0.0707	0.7490	0.0752	0.0826*	0.0728	0.0809	0.0795	0.0883	6.90
	Recall	0.1043	0.1088	0.1123	0.1221*	0.1073	0.1215	0.1161	0.1297	6.22
	MAP	0.0638	0.0649	0.0656	0.0757*	0.0596	0.0697	0.0675	0.0793	4.76
	NDCG	0.1215	0.1233	0.1234	0.1405*	0.1173	0.1320	0.1281	0.1457	3.70
Gowalla ($K = 64$)	Precision	0.0733	0.0788	0.0807	0.0873	0.0778	0.0895*	0.0866	0.0954	6.59
	Recall	0.1081	0.1147	0.1204	0.1301	0.1188	0.1331*	0.1268	0.1417	6.46
	MAP	0.0655	0.0711	0.0701	0.0811*	0.0655	0.0772	0.0761	0.0891	9.86
	NDCG	0.1241	0.1325	0.1315	0.1486*	0.1269	0.1444	0.1416	0.1602	7.80
ML-1M ($K = 16$)	Precision	0.1284	0.1810	0.2032	0.1285	0.1886	0.2092	0.2133*	0.2192	2.77
	Recall	0.0606	0.1008	0.1083	0.0607	0.0963	0.1130	0.1179*	0.1223	3.73
	MAP	0.0719	0.1070	0.1233	0.0720	0.1112	0.1281	0.1348*	0.1377	2.15
	NDCG	0.1416	0.2036	0.2258	0.1416	0.2081	0.2337	0.2415*	0.2472	2.36
ML-1M ($K = 64$)	Precision	0.1298	0.1861	0.2207	0.1314	0.2184	0.2237	0.2249*	0.2316	2.98
	Recall	0.0711	0.1059	0.1256	0.0715	0.1275	0.1263	0.1301*	0.1355	4.15
	MAP	0.0720	0.1119	0.1396	0.0937	0.1361	0.1432*	0.1427	0.1500	4.75
	NDCG	0.1445	0.2111	0.2491	0.1456	0.2470	0.2530	0.2539*	0.2638	3.90

a) Column 'Improve' indicates the percentage of improvement that GCC achieves relatives to the * results (higher is better).

(3) 所有方法特征向量维度 $K = 64$ 的实验结果要明显优于 $K = 16$ 的结果. 说明了适当增大特征向量维度能够捕获到更多用户和物品包含的特征因素, 进而使得各基准方法取得一定性能提升.

(4) BPRMF 在 4 个数据集的表现并不令人满意, 是因为该方法仅采用了最简单的矩阵分解捕获用户长期兴趣. 同时 FPMC 实验结果相较于 BPRMF 取得了显著提升, 说明了额外捕获用户短期兴

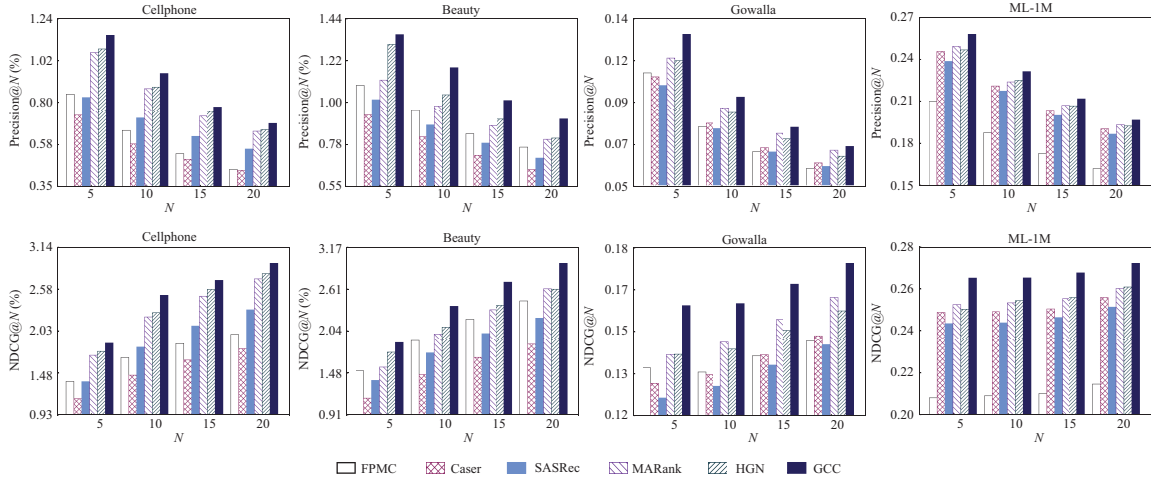


图 3 (网络版彩图) 不同推荐列表长度下各基准方法与 GCC 的实验结果对比 ($K = 64$)

Figure 3 (Color online) The results of baselines and GCC for different values of N ($K = 64$)

趣的做法是有效的。

(5) LightGCN 在不引入更多模型参数的前提下比 MF 取得更好成绩, 说明使用图卷积捕获用户长期依赖是有意义的. 同时 LightGCN 在 Gowalla 的部分结果优于 MARank 和 HGN, 是因为多层图卷积结构在面对蕴含丰富社交信息的数据时可以更有效地捕获协同过滤信号^[34], 从而取得更好表现.

(6) Caser 在 Gowalla 和 ML-1M 数据集的结果优于 FPMC, 但在 Cellphone 和 Beauty 上要远落后于 FPMC. 原因在于 Caser 为了实现 CNN 引入了大量模型参数, 因此在交互数据更丰富的稠密数据集可以更有效地捕获用户短期兴趣, 而在面对稀疏数据时则往往比较乏力. 对于结构更简单的 FPMC 则不需要大量的训练数据支撑即可取得不错表现. 此外, 对比 Caser 和 SASRec 也可得出相似结论.

(7) HGN 和 MARank 在 4 个数据集的结果显著优于 FPMC 和 SASRec, 原因包括以下两点: 其一, HGN 和 MARank 弥补了 FPMC 只能建模一阶马尔科夫链的局限性, 因此可以通过建模用户高阶时序信息更准确地刻画用户兴趣; 其二, FPMC 分解一阶马尔科夫链的做法, 以及 SASRec 的自注意力机制做法仅能建模物品个体层关系, 而 HGN 和 MARank 能够分别使用分级门网络和残差网络进一步捕获物品的联合层关系, 从而能够获取更丰富的用户兴趣信息.

(8) GCC 在所有指标下均取得了最佳表现. 在 $K = 64$ 时以 MAP@10 为例, GCC 分别取得了 12.93%, 10.50%, 9.86% 和 4.75% 的提升幅度. 原因包括以下 3 点: 其一, 所设计的个性化胶囊模块可以更有效地捕获物品联合层关系; 其二, 借助胶囊网络的空间感知能力可以建模更丰富的物品关系; 其三, 新的门单元结构可以在体现用户个性化差异的前提下更充分地捕获物品个体层关系. 另外, GCC 在稀疏数据集也取得了显著提升, 体现了 GCC 可以更有效应对稀疏性问题.

实验 2 模型简化实验分析

为了验证 GCC 中个性化胶囊模块和门单元模块各自的有效性, 本文设计了 3 个变种方法, 其中 CE-MF 对应 BPRMF 更改为二分类交叉熵损失函数后的实验结果. GCC-C 表示 GCC 移除门单元模块只保留个性化胶囊模块的实验结果. GCC-G 表示 GCC 移除个性化胶囊模块只保留门单元模块的实验结果. 为使各个组别的相对差距更加直观, Cellphone 和 Beauty 的结果均乘以 100%. 通过分析表 4 可以得出如下主要结论.

(1) GCC-G 比 CE-MF 取得更好结果说明了设计个性化门单元模块捕获个体层关系是有意义的.

表 4 模型简化实验结果

Table 4 Ablation study

Dataset	Metric@10	CE-MF	GCC-C	GCC-G	GCC
Cellphone (%)	Precision	0.6261	0.7845	0.9051	0.9534
	NDCG	1.6076	2.0654	2.3079	2.4887
Beauty (%)	Precision	0.6788	0.8471	1.1282	1.1822
	NDCG	1.4242	1.7520	2.2492	2.3689
Gowalla	Precision	0.0794	0.0949	0.0920	0.0954
	NDCG	0.1277	0.1583	0.1541	0.1602
ML-1M	Precision	0.1264	0.2232	0.2248	0.2316
	NDCG	0.1396	0.2539	0.2547	0.2638

(2) 对比 GCC-C 和 CE-MF, 前者取得了明显提升, 验证了采用胶囊网络建模物品联合层关系并保留局部有序性特征的做法是有效的. 此外, 综合对比表 3 中 Caser 在 4 个数据集上的实验结果, GCC-C 取得更佳表现说明了捕获最近物品局部有序性关系是有意义的, 同时一定程度上体现了胶囊网络比 CNN 更适用于建模用户高阶时序信息解决序列化推荐问题.

(3) CE-MF 在所有数据集的结果均远小于其他组, 证明了传统的只采用矩阵分解建模用户长期兴趣的做法无法取得令人满意的结果. 同时 GCC 在所有数据集中均取得了最佳结果, 进一步验证了门耦合胶囊网络能够更准确地刻画用户兴趣.

(4) 随着数据集稀疏性变化, GCC-C 的结果存在明显起伏. 因为稀疏数据集用户连续交互记录缺失值较多, 在连续的 L 个交互记录中, 相比稠密数据集相对缺乏丰富的联合层关系. 横向对比实验结果只能说明数据集本身存在差异, 不能判断特定模块的优劣.

实验 3 超参数敏感性分析

高阶时序信息阶数 L 的不同取值可以使 GCC 实际性能产生巨大起伏. 因此, 本文对超参数 L 分别取值 $\{2, 3, 4, 5, 6, 7\}$ 进行对比实验来研究其具体影响. 由于 Precision 和 NDCG 的实验结果图像有着相似的变化趋势, 受篇幅限制, 本实验只展示 NDCG@10 的相关结果, 如图 4 所示, 其中 “%” 表示实验结果乘以 100%. 通过分析可得以下主要结论.

(1) 在 4 个不同推荐场景中, 实验结果数值随 L 的增大呈现先增大后减小的趋势.

(2) 4 个数据集中 L 的最佳取值为 $\{3, 3, 3, 6\}$, 当 L 的取值大于最佳值后, 实验结果转为下降趋势. 例如在 Cellphone 数据集中, 当 $L > 3$ 时, 实验结果开始明显下降. 究其原因, 首先, 不同的数据集具有不同的主题, 意味着用户因特定兴趣产生的连续交互次数有限, 过大的序列阶数反而囊括了不同兴趣对应的交互行为. 其次, 数据集中连续的 L 次交互由于稀疏性或用户交互的随机性问题并不一定均同下一决策行为存在密切关联. 因此, 针对不同场景选择合适取值才能最优化性能表现.

实验 4 案例分析

为了进一步说明 GCC 对于局部有序性关系的建模能力, 本文在 ML-1M 数据集中随机选取用户 (真实编号为 10) 并截取该用户所属的一段长度为 6 ($L = 6$) 的连续交互序列进行实例测试, 具体如图 5(a) 中 $\{S_1, S_2, \dots, S_6\}$ 所示, 分别对应《当哈利遇到莎莉》、《大笨蛋》、《老板度假去》、《E.T. 外星人》、《鳄鱼邓迪》和《木乃伊》, 各个电影的核心题材类型均标注在相应编号中, 该段序列对应的下一时刻目标交互对象《狼踪》已在图中用红色标出, $\{S_A, S_B, \dots, S_G\}$ 指代不同模型预测的目标交互对象. 受篇幅限制, 本实验只展示最佳基准方法 HGN 以及 GCC 在推荐列表长度 $N = 3$ 时的推荐结果.

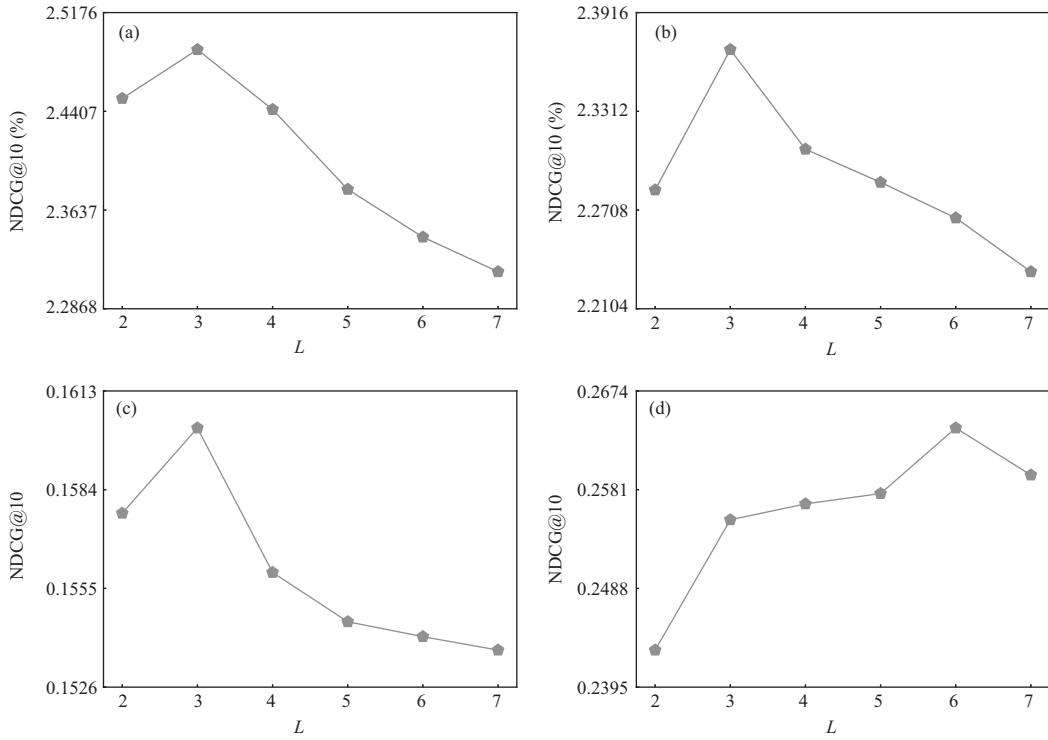


图 4 在 4 个数据集中不同 L 取值对应的 $NDCG@10$ 实验结果 ($K = 64$)

Figure 4 The results of $NDCG@10$ for different values of L on four real-world datasets ($K = 64$). (a) Cellphone; (b) Beauty; (c) Gowalla; (d) ML-1M

仔细分析图 5(a) 所示用户原始序列, 不难发现该用户虽然在 6 次交互中有 4 次均观看了有关“喜剧”题材的电影, 但随着时间推移交互历史中“喜剧”成分相对减少, 同时“冒险”因素不断加重, 因此该用户这段时间由于逐渐偏好“冒险”题材所以在下一时刻选择了冒险类电影. 带着这一分析我们将该序列输入已训练好的 HGN 和 GCC 模型中, 结果显示 HGN 和 GCC 在推荐列表中均能命中目标交互对象, 但 GCC 因其能够感知局部有序性关系从而准确刻画了用户兴趣的演变过程, 进而在第一个位置即实现准确推荐. 这一结果体现了感知用户交互序列中局部有序性关系的必要性.

之后, 调整上述序列中部分交互行为的相对次序, 如图 5(b) 所示. 分析该序列不难发现, 由于局部次序调整使得当前序列中“喜剧”和“冒险”成分均在不断增强, 显然此时用户兴趣解读为偏好喜剧和冒险题材的电影更为合适. 我们将该序列再次输入 HGN 和 GCC, 得到如图 5(b) 所示实验结果. 结果显示, HGN 由于对序列内部相对次序不敏感, 依然解读出同上次测试相同的兴趣偏好并输出相同的推荐列表. 而 GCC 则感知到因局部相对次序的变化对用户兴趣带来的影响, 进而做出与上次明显不同又合乎实际推理的推荐结果. 这一现象进一步说明了 GCC 捕获物品间局部有序性关系的有效性.

5 结论与展望

本文提出了一种新的序列化推荐方法——门耦合胶囊网络 GCC, 通过细粒度建模个体层、联合层以及局部有序性 3 个层面的物品间关系实现对用户兴趣的准确刻画. 并进一步论证了 GCC 可以被视为当前流行的序列化推荐方法的扩展或泛化. GCC 将胶囊网络应用于序列化推荐领域, 通过为物

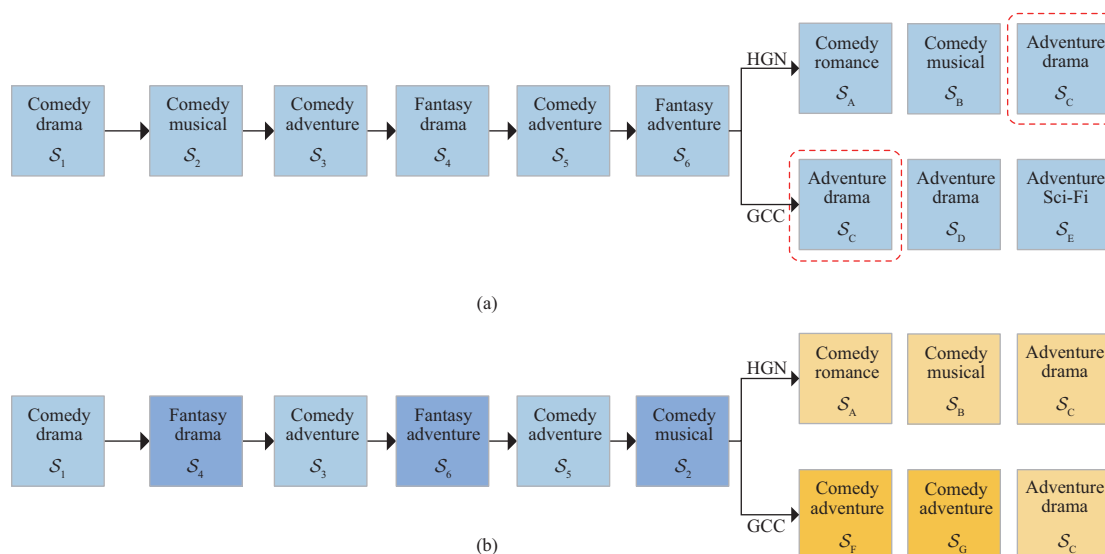


图 5 (网络版彩图) GCC 感知局部有序性关系实例描述

Figure 5 (Color online) The example description of GCC perceiving the local-order relationship. (a) Raw sequence; (b) adjustive sequence

品特征矩阵指定多个胶囊达到建模物品联合层关系的目的. 同时在此基础上借助胶囊网络的空间感知能力有效保留交互序列中的局部有序性特征. 另外, 新设计的门单元模块能够在满足用户个性化需求的同时捕获充分的个体层关系. 最后将用户长期兴趣同多类型短期序列特征有效融合. 本文在不同推荐场景的数据集上进行了大量实验, 结果表明 GCC 相对于当前流行的序列化推荐方法取得了显著性能提升, 实验结果也体现了在序列化推荐任务中推广胶囊网络的可行性.

由实验结果可知, 数据的稀疏性问题严重影响了推荐方法的性能表现. 最近, 知识图谱因其可以储存大量实体及实体间关系的特点受到人们广泛关注. 在未来的研究中, 我们考虑引入知识图谱建模辅助信息来缓解用户交互数据的稀疏问题和冷启动问题, 从而进一步提高序列化推荐方法的性能.

参考文献

- 1 Qin C, Zhu H S, Zhuang F Z, et al. A survey on knowledge graph-based recommender systems. *Sci Sin Inform*, 2020, 50: 937–956 [秦川, 祝恒书, 庄福振, 等. 基于知识图谱的推荐系统研究综述. *中国科学: 信息科学*, 2020, 50: 937–956]
- 2 Liu Q, Zhao H K, Wu L, et al. Illuminating recommendation by understanding the explicit item relations. *J Comput Sci Technol*, 2018, 33: 739–755
- 3 Rendle S, Freudenthaler C, Schmidt-Thieme L. Factorizing personalized Markov chains for next-basket recommendation. In: *Proceedings of the 19th International Conference on World Wide Web*. New York: ACM, 2010. 811–820
- 4 He R N, McAuley J. Fusing similarity models with Markov chains for sparse sequential recommendation. In: *Proceedings of the 16th IEEE International Conference on Data Mining*. Piscataway: IEEE, 2016. 191–200
- 5 Yu L, Zhang C X, Liang S S, et al. Multi-order attentive ranking model for sequential recommendation. In: *Proceedings of the 33th AAAI Conference on Artificial Intelligence*. Palo Alto: AAAI, 2019. 5709–5716
- 6 Tang J X, Wang K. Personalized top-n sequential recommendation via convolutional sequence embedding. In: *Proceedings of the 11th ACM International Conference on Web Search and Data Mining*. New York: ACM, 2018. 565–573
- 7 Yan A, Cheng S, Kang W C, et al. CosRec: 2D convolutional neural networks for sequential recommendation. In: *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. New York: ACM, 2019. 2173–2176
- 8 Ma C, Kang P, Liu X. Hierarchical gating networks for sequential recommendation. In: *Proceedings of the 25th ACM*

- SIGKDD Conference on Knowledge Discovery and Data Mining. New York: ACM, 2019. 825–833
- 9 Hidasi B, Karatzoglou A, Baltrunas L, et al. Session-based recommendations with recurrent neural networks. In: Proceedings of the 4th International Conference on Learning Representations. Puerto Rico: San Juan, 2016. 1–10
- 10 Hidasi B, Karatzoglou A. Recurrent neural networks with top-k gains for session-based recommendations. In: Proceedings of the 27th ACM International Conference on Information and Knowledge Management. New York: ACM, 2018. 843–852
- 11 Li Z, Zhao H K, Liu Q, et al. Learning from history and present: next-item recommendation via discriminatively exploiting user behaviors. In: Proceedings of the 24th ACM SIGKDD Conference on Knowledge Discovery and Data Mining. New York: ACM, 2018. 1734–1743
- 12 Wang H W, Guo M Y. Recurrent memory networks: modeling long short-term user preferences for session-based recommendation. *Sci Sin Inform*, 2020, 50: 1867–1881 [王鸿伟, 过敏意. 刻画长短期用户兴趣的基于会话的推荐系统. *中国科学: 信息科学*, 2020, 50: 1867–1881]
- 13 Kang W C, McAuley J. Self-attentive sequential recommendation. In: Proceedings of the 18th IEEE International Conference on Data Mining. Piscataway: IEEE, 2018. 197–206
- 14 Wu B, Lou Z Z, Ye Y D. Co-regularized matrix factorization recommendation algorithm (in Chinese). *J Software*, 2018, 29: 2681–2696 [吴宾, 娄铮铮, 叶阳东. 联合正则化的矩阵分解推荐算法. *软件学报*, 2018, 29: 2681–2696]
- 15 Rendle S, Freudenthaler C, Gantner Z, et al. BPR: Bayesian personalized ranking from implicit feedback. In: Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence. New York: AUAI, 2009. 452–461
- 16 Wang X, He X N, Wang M, et al. Neural graph collaborative filtering. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM, 2019. 165–174
- 17 He X N, Deng K, Wang X, et al. LightGCN: simplifying and powering graph convolution network for recommendation. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM, 2020. 639–648
- 18 Hinton G E, Krizhevsky A, Wang S D. Transforming auto-encoders. In: Proceedings of the 21st International Conference on Artificial Neural Networks. Berlin: Springer, 2011. 44–51
- 19 Sabour S, Frosst N, Hinton G E. Dynamic routing between capsules. In: Proceedings of the 31st Conference on Annual Conference on Neural Information Processing Systems. Cambridge: MIT Press, 2017. 3856–3866
- 20 Hinton G E, Sabour S, Frosst N. Matrix capsules with EM routing. In: Proceedings of the 6th International Conference on Learning Representations. British Columbia: Vancouver, 2018. 1–15
- 21 Chen B L, Huang X, Xiao L, et al. Hyperbolic capsule networks for multi-label classification. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. Stroudsburg: ACL, 2020. 3115–3124
- 22 Nguyen D Q, Vu T, Nguyen T D, et al. A capsule network-based embedding model for knowledge graph completion and search personalization. In: Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics. Stroudsburg: NAACL, 2019. 2180–2189
- 23 Lalonde R, Bagci U. Capsules for object segmentation. In: Proceedings of the 1st Conference on Medical Imaging with Deep Learning. Nijmegen: MIDL Foundation, 2018. 1–9
- 24 Li C, Liu Z Y, Wu M M, et al. Multi-interest network with dynamic routing for recommendation at Tmall. In: Proceedings of the 28th Conference on Information and Knowledge Management. New York: ACM, 2019. 2615–2623
- 25 Liang T T, Xia C Y, Yin Y Y, et al. Joint training capsule network for cold start recommendation. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM, 2020. 1769–1772
- 26 Dauphin Y N, Fan A, Auli M, et al. Language modeling with gated convolutional networks. In: Proceedings of the 34th International Conference on Machine Learning. New York: ACM, 2017. 933–941
- 27 Kingma D P, Ba J. Adam: a method for stochastic optimization. In: Proceedings of the 3rd International Conference on Learning Representations. California: San Diego, 2015. 1–15
- 28 He R N, McAuley J. Ups and downs: modeling the visual evolution of fashion trends with one-class collaborative filtering. In: Proceedings of the 25th International Conference on World Wide Web. New York: ACM, 2016. 507–517
- 29 Cho E, Myers S A, Leskovec J. Friendship and mobility: user movement in location-based social networks.

- In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2011. 1082–1090
- 30 Harper F M, Konstan J A. The movielens datasets: history and context. *ACM Trans Interact Intell Syst*, 2016, 5: 1–19
 - 31 Wu B, Ye Y D. BSPR: basket-sensitive personalized ranking for product recommendation. *Inf Sci*, 2020, 541: 185–206
 - 32 Wu B, He X N, Sun Z C, et al. ATM: an attentive translation model for next-item recommendation. *IEEE Trans Ind Inf*, 2020, 16: 1448–1459
 - 33 Sun Z C, Wu B, Wu Y D, et al. APL: adversarial pairwise learning for recommender systems. *Expert Syst Appl*, 2019, 118: 573–584
 - 34 Xu B B, Cen K T, Huang J J, et al. A survey on graph convolutional neural network (in Chinese). *Chin J Comput*, 2020, 43: 755–780 [徐冰冰, 岑科廷, 黄俊杰, 等. 图卷积神经网络综述. *计算机学报*, 2020, 43: 755–780]

Fine-grained modeling of user interests for sequential recommendation

Qi ZHANG, Bin WU, Zhongchuan SUN & Yangdong YE*

School of Computer and Artificial Intelligence, Zhengzhou University, Zhengzhou 450001, China

* Corresponding author. E-mail: yeyd@zzu.edu.cn

Abstract In recent years, the sequential recommendation method has attracted extensive attention owing to its practicability and high accuracy. Different from general recommendations, explicitly capturing short-term interests based on users' recent actions lies at the heart of a sequential recommender. Existing methods have either modeled successive items step by step without considering the many-to-one relationship or neglected the local-order information, considering the recent items as a set. This neither fully mines complex item transitions nor depicts the evolution of user interests. Therefore, we propose the gating coupled capsule network (GCC), a fine-grained analysis method to model individual-level, union-level, and local-order relationships of short-term user interests. Specifically, we designed a user-specific capsule module to capture high-level temporal interaction sequences, which model union-level patterns and perceive local-order relations. Moreover, we present a personalized gating module to focus on the pairwise relationships among items to capture the influence of individual-level information on short-term user interests. Extensive experiments under four real-world recommendation scenarios demonstrate that GCC outperforms the state-of-the-art methods on different ranking metrics.

Keywords capsule networks, sequential recommendation, gating mechanism, implicit feedback, recommender systems