arXiv:2303.02851v1 [cs.IR] 6 Mar 2023

神经推荐系统增量更新综述

PEIYAN ZHANG, 香港科技大学，香港SUNGHUN KIM, 香港科技大学，香港

推荐系统（RS）旨在为用户推荐物品，缓解消费者选择过载问题。尽管研究人员已对推荐系统的不同方面和挑战开展了大量研究，但学术研究与工业应用之间仍存在差距。具体而言，现有的大多数模型仍以离线方式运行，即在大型静态训练集上训练推荐系统，并通过一次性流程在非常受限的测试集上进行评估。推荐系统会一直保持不变，直到进行下一批次的重新训练。我们将此类推荐系统称为批量更新推荐系统（BURS）。在现实中，推荐系统必须面对这样的挑战：随着新数据流不断涌入，系统需要即时更新，并基于新到达的数据，针对当前的用户活动生成更新后的推荐结果。我们将此类推荐系统称为增量更新推荐系统（IURS）。

本文对神经推荐系统的增量更新进行了系统综述。我们首先介绍关键概念，并对 IURS 任务进行形式化定义。随后，我们阐述 IURS 相较于传统 BURS 所面临的挑战。之后，我们详细介绍现有文献和评估问题。最后，我们概述该领域一些重要的开放研究问题。

CCS 概念： • **信息系统** → **推荐系统**。

附加关键词和短语：增量更新、批量更新、推荐系统、概念漂移

ACM 引用格式：
Peiyan Zhang 和 Sunghun Kim。2023年。《神经推荐系统增量更新综述》。《*ACM*学报》37卷，4期，文章 111（2023年8月），18页。https://doi.org/XXXXXXX.XXXXXXX

1 引言

推荐系统（RS）[76] 旨在帮助用户应对消费选择过多的问题，为其提供个性化的物品推荐。针对推荐系统的不同方面和挑战，研究人员已开展了广泛研究。一般而言，生成推荐结果时可以考虑多种因素。例如，用户交互序列、物品特征和用户特征等侧信息，以及时间戳 [75] 和空间位置 [69], 等。根据输入数据的不同，推荐模型可分为基于内容的推荐系统、协同过滤和混合推荐系统 [1]。

随着深度学习的发展，当前推荐系统研究中，深度学习技术被广泛用于开发更强大的神经推荐模型。在学术界，Hidasi等人[47] 率先提出利用循环神经网络（RNNs）对用户偏好进行建模，之后又通过专为推荐任务设计的排序损失函数加以改进 [46]。随后，基于注意力的机制被引入系统，显著提升了性能。NARM[55] 在循环神经网络模型上运用注意力机制，以增强

作者地址：Peiyan Zhang，pzhangao@cse.ust.hk，香港科技大学，香港；Sunghun Kim，hunkim@ cse.ust.hk，香港科技大学，香港。

允许免费为个人或课堂用途制作本作品全部或部分内容的数字或纸质副本，前提是不得为营利或商业利益而制作或分发副本，且副本首页须附上本声明及完整引文。本作品中归 ACM 以外其他方所有的组件，其版权必须得到尊重。注明出处后可以摘录。以其他方式复制、再版、发布到服务器或分发至列表，均须事先获得明确许可和/或支付费用。请通过 permissions@acm.org 申请许可。© 2023年 计算机协会。提交给 ACM 的稿件

STAMP 捕获特征时 [58] 通过一个简单的注意力模型捕获长期和短期偏好。卷积神经网络（CNN）也被用于基于会话的推荐。Tang 等人 [83] 尝试将物品会话嵌入为矩阵，并对矩阵执行卷积以获得表示。Yuan 等人 [105]进一步通过空洞卷积层改进了该方法，从而扩大感受野。近来，更多研究着眼于利用图神经网络（GNN） [17, 42, 71, 81, 90, 95, 98, 102]探索会话数据中的复杂物品转移。例如，Wu 等人 [98] 首先提出利用图结构捕获复杂转移。随后，Pan 等人 [71] 尝试通过高速公路网络避免过拟合 [81]。此外，Zhang 等人通过实证识别出基于图神经网络的模型存在冗余， [107] 移除了图神经网络传播部分，并提出一种更轻量但有效的模型。该模型具备多层次推理能力，能够更准确、高效地捕获用户偏好。

在工业界，神经推荐模型也广泛应用于大多数电子商务和内容系统中，例如亚马逊、Netflix、YouTube 和 Google 新闻 [19, 22, 24, 38, 68]。例如，亚马逊和阿里巴巴 [114] 已部署了各种推荐系统，每天处理数十亿次页面浏览量，并根据顾客的偏好识别并推荐他们可能购买的商品。YouTube 的 [24] 首页推荐占视频点击量的 60%，Netflix 的[38] 电影推荐占 80%。随着每天涌现出海量新闻文章，MSN 新闻 [96] 利用新闻推荐系统捕捉用户兴趣并生成个性化新闻推荐，帮助用户高效找到感兴趣的新闻文章，缓解信息过载。

尽管如今神经推荐系统已取得巨大成功，但大多数现有模型仍以离线方式 [66], 运行：推荐模型在一个非常庞大的静态训练集上训练，并在一个规模非常有限的测试集上进行一次性评估。推荐系统会保持不变，直到执行下一轮批量重新训练。我们将此类推荐系统称为批量更新推荐系统（BURS）。现实中，它们必须面对这样的挑战：推荐系统需要随着新数据流入而即时更新，并基于新到达数据为当前用户活动生成令人满意的推荐结果。例如，用户点击某个物品后，这一行为应立即反映在推荐系统中。然而，要确保良好推荐质量的推荐模型往往会遭遇极高的延迟。定期使用整个数据集重新训练模型也不可行。海量参数给计算资源和数据存储带来了严峻挑战。此外，如果我们仅使用新收集的数据重新训练模型，那么来自新用户的数据量较少，将难以生成令人满意的推荐结果。

具体而言，BURS 有两个主要局限。首先，模型更新完成后，推荐系统无法考虑此后发生的用户交互，直到下一次模型更新，因此无法将持续的用户活动纳入考量。尽管用户偏好和项目感知会随时间不断变化，系统仍使用同一个模型来提供推荐结果。在这种情况下，最新观察数据无法立即纳入考量，导致推荐质量不佳 [27]。其次，随着数据集不断增大，周期性再训练的计算成本也会变得高昂。因此，这会带来可扩展性问题。

推荐系统领域的研究主要关注了其中一部分局限性，并分别采用不同方法加以解决，即时间感知推荐系统 [13] 和持续学习推荐系统 [66]。然而，这些解决方案无法克服前面提到的两项局限性，因为它们并非旨在同时处理新观测数据和建模时间动态。近期研究 [72, 109] 尝试将推荐问题表述为增量更新问题，并通过设计高效的增量更新推荐系统（IURS）来解决；这类推荐系统只需利用新到达的数据即可即时更新，从而捕捉用户兴趣漂移以及物品受欢迎程度的变化。

IURS 能够实时响应变化，且其运行不受观测数据数量的影响，从而解决了前文所述 BURS 的局限性。

总的来说，在动态场景下，IURS 比传统的 BURS 更贴近现实。尽管现有研究（即本综述所回顾的工作）探讨了 IURS 的多个方面，但该领域的研究仍较为零散，一些重要问题仍未得到解答。本综述将回顾 IURS 领域的现有研究，讨论其与其他推荐系统类别的差异，并指出现有方法的缺陷。

本综述其余部分的安排如下：我们首先在第 2 节介绍一些背景知识，并对 IURS 任务进行形式化定义。然后，在第 3 节阐述 IURS 相较于传统批量推荐系统所面临的挑战。接下来，我们分别在第 4 节和第 5 节详细介绍现有文献和评估问题。最后，我们在第6 节讨论值得关注的开放研究问题，并在第 7 节总结本文。

2 问题定义

本节首先介绍 BURS 的经典场景，以便与 IURS 的流式场景进行对比。随后，我们阐述 IURS 的流式场景。最后，为了清晰界定所讨论的问题，我们详细描述 BURS 和 IURS 的算法。

2.1 BURS 的经典设置

推荐系统旨在为用户推荐物品的个性化建议，以应对消费者选择过载问题。针对这一主题的研究需要对该问题进行形式化描述，以便量化评估。第一种形式称为评分预测，重点是预测用户倾向于对尚未评分的物品给出的评分值。第二种形式是 Top-K 推荐，旨在推荐最符合用户当前偏好的 Top-K 物品列表。预期推荐系统的核心在这两种情况下都应遵循相同的流程 [1]。也就是说，推荐模型会预测每件物品的效用，即评分预测中的评分分数，或 Top-K 推荐中的相关性分数。推荐物品的选择基于效用最大化。

以往的推荐系统开发主要侧重于批处理场景，即整个训练数据集均可用且保持静态。推荐模型会通过一次性流程进行评估，并保持静态，直至需要重新训练。此类做法与实际应用之间存在巨大差距：在实际应用中，推荐模型应能随着新数据流的涌入而即时更新，并根据新到达的数据，针对当前用户活动生成令人满意的推荐结果。

在静态领域中对用户-物品关系进行建模时，由于可用数据充足且相关性稳定，因此无需实现能够在新交互生成后立即将其纳入考虑的 IURS。然而，在真实的动态领域中，用户兴趣会发生漂移，物品受欢迎程度也可能变化，因此及时考虑以高频率到达的新交互非常重要，这就需要 IURS。

2.2 IURS 的流式场景

流式场景。随着数据持续不断地流入，且没有明确的终止时间，分析、计算和存储方面出现了许多新的研究挑战。具体而言，与传统 BURS 相比，以下差异尤为突出 [36]：

- 新的交互持续到达。
- 数据流可能是无界的。

- 一旦某个观测数据从数据流中被处理，它要么被丢弃，要么保留在一个相对于数据流大小而言较小的记忆库中。只有当观测数据被保留在记忆库中时，才能被检索。

2.3 算法描述

为了更清楚地理解这个问题，我们在此描述 IURS 所遵循的算法流程，并将其与传统的批量推荐系统流程进行对比。

算法1 BURS 算法

输入： D^0：初始观测交互集； T：模型重新训练时间周期；

$D \leftarrow D^0$;
$M \leftarrow f_{train}(D)$;
函数推荐
  对每个用户 u 在时间 t **执行**
    $list = f_{predict}(M, u)$;
    观察用户 u对物品 i的反馈 r_{ui};
    $D \leftarrow D \cup \{u, r_{ui}, t_{ui}, i\}$;
function重新训练
  对于 每个时间周期 T **执行**
    $M \leftarrow f_{train}(D)$;
推荐() & 重新训练();

算法2 IURS算法

输入： D^0：初始观测交互集； T：模型重新训练时间周期；

$D \leftarrow D^0$;
$M \leftarrow f_{train}(D)$;
$D_{mem} \leftarrow f_{update}(D)$;
函数 训练(M, u, r_{ui}, t_{ui}, i, D_{mem})
  $M \leftarrow f_{incre-train}(M, u, r_{ui}, t_{ui}, i, D_{mem})$;
函数推荐
  对每位用户 u 在时间 t **执行**
    $list = f_{predict}(M, u)$;
    观察用户 u对物品 i的反馈 r_{ui};
    训练(M, u, r_{ui}, t_{ui}, i, D_{mem});
    $D_{mem} \leftarrow f_{update}(u, r_{ui}, t_{ui}, i)$;
推荐();

符号说明。 为说明这些算法，我们首先定义一组符号。令 $U = \{u_1, u_2, \ldots\ldots, u_{|U|}\}$ 为系统处理的所有用户构成的用户全集，$I = \{i_1, i_2, \ldots\ldots, i_{|I|}\}$ 为物品全集。我们将 D 记为观测交互集，其中每个交互由 (u, r_{ui}, t_{ui}, i) 组成。r_{ui} 是用户 u对物品 i 的反馈，而 t_{ui} 表示交互发生时的时间戳。需要注意的是，用户可能会对物品产生多种反馈，例如浏览、点击、评分、购买等。由于本综述不探讨反馈类型，因此这里不使用额外符号记录反馈类型。提交给 ACM 的稿件

根据上述符号，BURS 和 IURS 算法的流程分别如算法 1 和算法 2所示。对于 BURS，推荐模型会为用户 u 生成预测物品列表（算法 1，第 5 行）。一旦观测到用户 u对物品 i的反馈，该交互就会被添加到整个数据集中。然而，在下一次重新训练之前，模型不会更新。在一定的时间周期 T后，模型将使用整个数据集进行重新训练 D（算法 1，第 10 行），这在实际应用中难以扩展。相比之下，对于 IURS，每当推荐模型为用户生成推荐列表 u（算法 2，第 8 行）并观测到用户 u的反馈（算法 2，第 9 行）时，模型都会利用记忆库中的数据进行增量训练（算法 2，第 10 行）。随后，根据新观测交互更新数据记忆库（算法 2，第 11 行）。

3 挑战

IURS 与传统 BURS 之间的差异带来了若干挑战，增量更新推荐系统应妥善应对这些挑战。本节将对以下挑战进行介绍。

3.1 系统动态建模

在动态在线推荐系统中，用户兴趣可能会显著漂移，物品流行度也会随时间发生变化。因此，准确地对系统动态进行建模变得十分迫切，因为许多底层对象都在以不同的速率、不同的方式演化。BURS 在两次重新训练之间将保持不变，并提供相同的推荐结果，无法即时考虑最新观测，从而导致推荐质量较差。IURS 需要应对这种动态场景，并适当地建模时间相关性。

3.2 增量学习

对 IURS 进行增量更新是一项长期工作。为了处理持续时间较长的 IURS，以往的方法尝试利用大型记忆库 [65] 学习整个历史数据的表示，或保留代表性样本以便一段时间后重新训练[74]。然而，由于数据会无限持续流入，记忆库容量有限，无法保留所有代表性样本。因此，IURS 应避免采用从头训练范式，转而进行增量学习。要实现 IURS 的增量学习，我们还必须解决两项具有挑战性的任务。一是灾难性遗忘：随着增量学习的新知识不断覆盖过去学到的知识，模型可能会遗忘先前学到的知识。另一项是有效迁移对未来推荐有用的过去知识。

4 相关工作

4.1 系统动态建模

在 IURS 中，用户兴趣可能发生显著漂移，而物品流行度也会随时间变化。因此，将时间信息纳入考量十分重要。已有研究提出了若干将时间融入推荐系统的方法 [13]。我们主要将融合时间与推荐系统的方法分为三类。

4.1.1 基于新近性的时间感知模型。基于新近性的时间感知模型假设，近期观察到的信息比更早的信息更相关，因为它们更能代表实际情况；因此，在学习和推荐过程中应给予它们更高的权重。请注意，此处的时间戳仅表示序列中的物品顺序或位置编号。大多数基于序列和基于会话的推荐模型都遵循这一思路。它们倾向于根据最近几个物品生成短期偏好，并将该表示与

长期偏好表示[42, 45, 71, 81, 90, 95, 98, 102]。请注意，与下文介绍的时间感知模型相比，基于新近性的时间感知模型相当简单，却能有效捕捉用户兴趣漂移。不过，这些模型大多在静态设置下运行。

4.1.2 上下文时间感知模型。 上下文时间感知模型将时间信息视为上下文维度，从而整合时间维度。Fan 等人 [32] 统一了序列模式和时间协同信号。为捕捉时间效应，他们将用户 - 物品嵌入与交互时间嵌入进行拼接。为生成交互时间嵌入，他们将时间跨度（时间相关性）建模为对应编码时间嵌入的点积。因此，时间编码函数可以推广到任意未见过的时间戳，从而能够通过编码时间嵌入推断出训练数据中未出现的任何时间跨度。借助生成的时间嵌入，他们能够探索时间效应并挖掘数据中的序列模式。该方法在 Top-K 推荐场景下的五个数据集上进行了评估。与其他时间感知方法（包括 TiSASRec[56]）相比，所提出的方法在召回率和平均倒数排名指标方面表现更佳。

为刻画用户侧和物品侧的动态变化，Chen 等人 [18] 提出为每个时间切片构建一个全局用户-物品交互图，并利用时间切片图神经网络学习用户和物品表示。然而，在每个时间切片中，他们将这些图视为静态图，并沿图聚合信息。为体现时间属性，他们分别使用两个门控循环单元，从用户侧和物品侧沿所有时间切片聚合用户/物品表示。这种系统建模方法本质上仍是静态的。

采用时间戳对时间衰减的影响进行建模，是另一种常见做法。Zhang 等人 [106] 构建了一个分层注意力网络，并在注意力权重上加入时间衰减因子，以反映动态性。这一思路基于这样一种观察：人们往往会在短时间内阅读内容相似的新闻。然而，经过较长时间后，这些新闻对用户的影响会减弱。Li 等人 [57] 将时间相关性编码到一个产品-产品图中，并根据用户在时间戳 t 之前购买的产品集合，利用时间衰减函数计算用户嵌入。时间衰减函数确保产品购买时间越晚，其权重越大。受强化学习中记忆衰减的启发，Zhou 等人 [115] 使用指数分母重新缩放边权重。因此，随着时间差增大，历史边权重的影响会逐渐减弱，从而对动态系统中的边衰减进行建模。

4.1.3 连续时间感知模型。连续时间感知模型将用户交互表示为时间的连续函数，并通过数据学习模型参数。

Bai 等人 [8] 提出了一种需求感知的霍克斯过程 [54]，用于捕捉过往物品对未来需求的复杂影响。由于用户需求对时间高度敏感，因此构建此类连续时间模型十分自然：需求可能由用户近期的购买行为产生，也可能由一段时间之前的购买行为产生，即短期需求和长期需求。之后，他们通过卷积式时间感知 LSTM 对短期需求建模，同时通过自注意力组件捕捉长期需求。

Kumar 等人 [53] 采用耦合循环神经网络，联合建模用户和物品的演化，并预测用户未来的嵌入轨迹。此外，他们提出了嵌入投影操作，用于投影用户在未来某个时间的嵌入。他们通过哈达玛积将时间纳入投影后的嵌入。沿着这种方式，随着时间推移，预测的用户嵌入会逐渐偏移得更远。当观测到下一次交互时，

嵌入会再次更新。请注意，这种方法采用完全连续的方式，即每次交互都会触发模型更新。

Choi 等人 [21] 首先将神经常微分方程（NODEs）引入连续推荐系统建模。通过重新设计 NODEs 框架下的线性图卷积网络 [44] ，他们成功学习出最优架构，而非依赖人工设计的架构；同时还能学习平滑的常微分方程解，这被认为适用于协同过滤。近期，Guo 和 Zhang 等人 [40] 提出了图嵌套GRU常微分方程（GNG-ODE），进一步将 NODEs 的思想扩展到连续时间时序会话图。GNG-ODE 能够保留动态用户偏好的连续性，将物品转移的时间和结构模式编码为连续时间动态嵌入表示，从而捕捉细粒度的用户偏好演化。

4.2 增量学习

要实现 IURS 的增量学习，我们还必须解决两个具有挑战性的任务。一是灾难性遗忘，即模型可能会忘记先前学到的知识，因为增量学习的新知识最终会覆盖过去学到的知识。另一个任务是有效迁移对未来推荐有用的过去知识。

事实上，遗忘问题并非推荐系统增量更新所独有，而是在使用来自不同分布的数据更新神经网络时普遍存在 [52]。持续学习是一个旨在解决灾难性遗忘问题、实现终身学习的研究方向，已成功应用于计算机视觉 [26] 和自然语言处理 [10]。然而，对于推荐系统增量更新，直接采用这些方法可能无法取得理想效果。这是因为两者的最终目标不同：持续学习旨在提升当前任务的表现，同时不牺牲先前任务的性能；而推荐系统的增量更新只关注未来推荐结果的性能 [109]。

近期，一些研究尝试将持续学习与推荐系统相结合，希望训练出一种稳健的模型，其中包含全部历史推荐经验，以促进新的推荐任务。具体而言，目前已提出以下两类研究工作。

4.2.1 基于样本的方法。第一类研究采用基于样本的方法，即存储一小组来自先前数据、具有代表性的序列，也就是样本集。具体来说，这类方法通常包含两个部分：样本集更新和表示更新。

样本集更新。每个物品的样本集元素称为该物品的样本，即包含与该物品交互的用户交互序列。出现新物品 i 后，基于样本的方法会平衡每个物品的样本集大小。例如，现有物品的样本集大小会减少，同时会为新物品 i创建一个新的样本集。假设由于记忆库容量限制，样本集总大小为 M，那么每个物品可获得 $m = \frac{M}{|I|}$的样本配额，其中 $|I|$ 是现有物品集合的大小。此外，每个物品的样本配额也可以与该物品的出现频率成正比。下面说明如何选择样本。样本的平均特征向量应接近对应物品的平均特征向量。因此，可以选择最具代表性的样本，从而保留该物品的一般属性。

样本集更新过程如算法 3所示。对于每个现有物品，我们首先根据包含该物品的交互序列计算其平均特征向量（算法 3，第 2 行）。之后，我们根据与物品平均特征向量的距离选择 m个样本（算法 3，第 3-4 行）。注意，这些 m 个样本

根据该距离进行排序，这有助于在未来出现新物品时减小样本集大小。最终，我们可以为每个现有物品获取更新的样本集（算法3，第 6 行）。请注意，我们会丢弃所有不属于样本集的交互。

算法3 样本集更新

输入： $S_i = \{s_1、s_2、\ldots\ldots、s_n\}$：包含物品 i 的交互序列；m：样本集大小；ϕ：基于输入交互序列计算序列表征的特征提取器；

输出： P_t：时间 t 时所有物品的样本集；

1: **对于** y，取 1，……，$|I_t|$ **执行**
2: $\mu \leftarrow \frac{1}{n}\Sigma_{s \in S_y}\phi(s)$;
3: **对于** k，取 1 ,...,m **do**
4: $p_k \leftarrow argmin_{s \in S \&\& s \notin \{p_1,\ldots,p_{k-1}\}} ||\mu - \frac{1}{k}[\phi(s) + \Sigma_{j=1}^{k-1}\phi(p_j)]||$;
5: $P_{ti} \leftarrow (p_1, p2, ..., p_m)$;
6: $P_t \leftarrow (P_{t1}, P_{t2}, ..., P_{t|I|})$;
7: **返回** P_t;

表示更新。完成当前数据集的样本集构建后，基于样本的方法将执行表示更新操作。算法4 详细说明了更新特征表示的步骤。给定时间 t 的新观测交互以及时间 P_{t-1} 的样本集 $t-1$，我们首先使用时间 $t-1$ 的模型参数初始化模型（算法4，第1行），并在数据 $D_t \cup P_{t-1}$ 上训练模型（算法4，第2–3行）。之后，我们使用算法 t 3更新时间 [48]的样本集大小。通常在训练过程中，会采用知识蒸馏来防止模型遗忘先前知识。

算法4表示更新

输入： D_t：时间 t 的新观测交互；P_{t-1}：时间 $t-1$ 的样本集；θ_{t-1}：时间 $t-1$ 的模型参数；

输出： θ_t：时间 t 的更新后的模型参数；P_t：时间 t 的更新后的样本集；

1：使用 θ_{t-1}初始化 θ_t；
2：**当** θ_t 未收敛时**执行**
3: 在数据 $D_t \cup P_{t-1}$上训练 θ_t；
4: 使用算法 P_t 3计算;
5: **返回** θ_t，P_t；

遵循这一思路，[29] 会维护一个样本库来保存历史信息，并随机选取样本。[16, 41,65, 66, 74, 93, 112] 也会维护样本库来保存历史信息。不过，它们会根据精心设计的启发式方法选取样本。基于样本的方法可以看作增量更新与使用完整历史数据进行训练之间的一种折中，旨在平衡短期利益与长期记忆 [72]。

尽管基于样本的方法在公共数据集上取得了巨大成功，但这类方法在实践中并不适用。一方面，样本库／记忆库的容量有限，而流式数据却是无限的。即使经过仔细筛选，选出的样本具有一定的信息量，它们仍不足以全面反映数据分布。另一方面，大多数这类方法会直接重放存储在样本库中的样本，而不会判断这些数据对当前任务是否有用。

4.2.2 基于模型的方法。 另一类工作称为基于模型的方法，旨在通过将过去的知识迁移到当前模型中来防止灾难性遗忘。该方法将过去的模型视为教师

模型，Wang 等人 [94] 使用知识蒸馏损失对学习过程进行正则化，以防止灾难性遗忘。此外，后续研究还将蒸馏技术用于图神经网络 [99] 和基于会话的推荐系统 [66]。然而，需要注意的是，仅仅限制网络的变化并不能将有用的知识迁移到当前任务中。换言之，过去模型中的某些知识可能不适用于当前任务。Zhang 等人 [109] 提出了一种针对下一时期数据进行显式优化的方法，大大缓解了 [66, 94, 99]问题。为了优化模型在未来的应用，该方法开发了一个迁移模块，用于整合不同时期模型的知识。然而，该方法未能考虑长期序列模式，因为模型主要关注两个连续时期之间的迁移。Peng 等人 [72] 首先针对每个时期的数据训练模型。该方法设计了一个超网络模型，将先前模型的参数作为输入，并输出一个经过微调的模型，以用于当前任务。然而，它要求超网络模型对推荐模型的每个参数进行微调；对于参数量巨大的大规模推荐模型而言，这并不可行。

5 评估

本节回顾 IURS 的评估方法和指标。与传统的 BURS 不同，IURS 会考虑交互之间的时间相关性，并对用户兴趣和物品流行度等多个不断演变的概念进行建模，因此需要采用不同的评估框架。

众所周知，推荐系统有在线评估和离线评估两种方式 [39]。然而，本节着重回顾离线评估，因为在线评估需要在线环境来为真实用户提供推荐，这一点难以实现。尽管如此，在线评估仍是一个值得探索的重要领域，因为它能够为推荐系统的最终目标提供最有说服力的证据。

5.1 BURS 的离线评估

由于离线评估无需与真实用户交互，且在复现和比较不同方法方面效率很高，因此它是推荐系统评估流程中最广泛采用的设置之一 [39]。

基础的离线评估框架基于广泛使用的训练集—测试集划分以及交叉验证流程。给定观测到的数据集后，会将其划分为训练集和测试集，前者用于训练模型，后者用于评估模型的性能。实现上述数据集划分有多种策略，例如随机划分、按时间划分和留一法策略 [39]。

尽管传统的批量评估采用留出法，且广泛用于评估推荐系统，但在数据流环境中仍面临若干挑战 [87]。如果将数据随机划分为训练集和测试集，就会忽视甚至破坏时间维度上的语义。此外，打乱数据可能导致不合逻辑的操作，例如使用未来的观测数据来预测过去的观测数据，从而造成严重的数据泄漏问题。从打乱后的观测数据中学习，会对捕捉交互之间真实的时间相关性产生负面影响。此外，传统的批量评估假设模型在生成推荐结果时保持静态。然而，只要生成推荐结果，IURS 就会持续更新。

5.2 IURS 离线评估

与传统的 BURS 评估相比，数据流设置给 IURS 的恰当评估带来了新的挑战。为解决这一问题，人们提出了多种尝试 [62, 70, 78, 87] ，其中大多数依赖于所谓的前序评估方法 [37]。

5.2.1 评估方法。与传统 BURS 的一次性过程评估不同，IURS 预期采用前序评估。在前序评估中，每次观察到用户反馈后，都会执行额外的先测试后学习过程 [87]。以 Top-K 推荐为例：

(i) 预测：为用户 u生成预测的 Top-K 物品列表。(ii) 评估：观察到用户 u的真实交互后，对预测物品列表进行评估。(iii) 更新：利用用户 u 的真实交互更新推荐模型。

前序评估方法具有多项优势。除了遵循已观察到的交互之间的时间关系外，它还能持续追踪推荐系统性能以及系统随时间的演化。同时，推荐模型还可以利用前序评估获得的实时统计数据，更好地进行用户兴趣漂移检测。然而，作为一种离线评估方法，它也并非没有缺点。其主要局限在于，预测物品列表仅根据一个物品进行评估，无法考察列表中出现的其他潜在候选项。不过，采用混合评估方法可以在一定程度上缓解这一局限。Siddiqui 等人 [78] 提出，基于一部分数据采用混合留出评估，并基于其余数据采用前序评估。

IURS 在线部署提供服务之前，仍需要先进行离线预热训练阶段。因此，实际应用环境并非完全处于流式设置中。因此，离线评估也应模拟这种工作流程，即考虑从批处理模式切换到流式模式。为应对这一挑战，Matuszyk 等人提出了以下评估协议 [62], ，将评估阶段分为三个阶段：

(1) 离线预热训练阶段。在此阶段，约前 30% 的数据被划分出来，以批处理模式训练模型。此阶段使用的数据作为训练集。(2) 批量测试—流式训练阶段。在此阶段，使用接下来的约 20% 数据测试在离线预热训练阶段训练的模型。请注意，测试后，这些数据也用于增量更新模型。此阶段使用的数据也可作为验证集。(3) 流式测试—流式训练阶段。在此阶段，利用剩余数据进行前序评估 [87]。

5.2.2 评估指标。前序评估指标基于每条观测到的用户反馈进行评估。我们可以了解性能随时间的演化情况。整体性能将通过对所有观测到的反馈取平均值来报告。以下介绍前序评估中广泛使用的一些指标[34]。请注意，$rank_u(i)$表示物品 i 在为用户 u生成的 Top-K 物品列表中的排名位置。

- 精确率分数衡量测试过程中已更正的推荐项所占的比例：

$$Precision@K(u,i) = \begin{cases} 0 & \text{if } rank_u(i) > K \\ \frac{1}{K} & \text{otherwise.} \end{cases} \tag{1}$$

表 1. 用于评估 IURS 的公共数据集

数据集	用户	物品	事件	领域	参考文献	论文中使用	
亚马逊图书	2.5M	929k	12.8M	Book	[64]	[21, 34]	
亚马逊电子产品	884k	96K	1.3M	电子产品	[64]	[34]	
亚马逊电影	1.3M	235k	8.5M	电影	[64]	[34]	
约会	194k	194k	11.7M	社交网络	[11]	[82, 89]	
Delicious	1.8k	69k	105k	兴趣点	[12]	[59]	
Diginetica	780k	43k	983k	电子商务	[98]	[66, 115]	
豆瓣	36k	726k	5.3M	社交网络	[113]	[59]	
Flixster	147k	48k	8.1M	电影	[30]	[63]	
Globo	314k	46k	3M	News	[25]	[25, 35]	
Gowalla	30k	41k	1M	兴趣点	[20]	[21]	
京东	457k	55k	8.9M	电子商务	[110, 111]	[8]	
Last.fm-600k	164	65k	493k	音乐	[7]	[86–88]	
Last.fm-30M	40k	5.6M	31M	音乐	[85]	[34, 53]	
Lazada	10k	44k	6.7M	电子商务	[72]	[72]	
MovieLens-100k	1k	1.7k	100k	电影	[43]	[5, 15, 63, 89]	
MovieLens-1M	6k	4k	1M	电影	[43]	[3, 63, 78, 86–88, 93]	
MovieLens-10M	72k	10k	10M	电影	[43]	[3, 15, 34, 80]	
MovieLens-20M	138k	27k	20M	电影	[43]	[72, 92]	
Netflix	480k	17k	100M	电影	[15]	[15, 63, 80, 92, 93]	
Plista	220k	835	1.1M	News	[51]	[5, 50, 60]	
Reddit 帖子	10k	1k	672k	用户帖子	[53]	[53]	
Sobazaar	17k	25k	843k	电子商务	[72]	[72]	
大丰	26k	24k	818k	电子商务	[91]	[8]	
淘宝	26k	24k	818k	电子商务	[117–119]	[8]	
天猫	50k	44k	3M	电子商务	[72]	[72]	
Twitter	413k	37k	35M	用户帖子	[100]	[28, 29]	
维基百科编辑	8k	1k	157k	用户帖子	[53]	[53]	
Yahoo! Music-6k	6k	127k	476k	音乐	[88]	[88]	
Yelp2018	32k	38k	1.6M	兴趣点	[44]	[21]	

- 召回率分数表示 top-k 推荐列表是否包含目标项目：

$$Recall@K(u,i) = \begin{cases} 0 & \text{if } rank_u(i) > K \\ 1 & \text{otherwise.} \end{cases} \tag{2}$$

- 折损累积增益 *(DCG)* 得分用于衡量排名质量：

$$DCG@K(u,i) = \begin{cases} 0 & \text{if } rank_u(i) > K \\ \frac{1}{\log_2(rank_u(i)+1)} & \text{otherwise.} \end{cases} \tag{3}$$

- 平均倒数排名（*MRR*）分数用于衡量目标项目在 top-k 推荐列表中的位置。MRR@K 分数越高，意味着目标项目在 top-k 推荐列表中的排名越靠前。如果目标项目不在推荐列表中，MRR@K 分数将设为零：

$$MRR@K(u,i) = \begin{cases} 0 & \text{if } rank_u(i) > K \\ \frac{1}{rank_u(i)} & \text{otherwise.} \end{cases} \tag{4}$$

5.2.3 数据集。当前推荐系统领域的研究也见证了公共数据集的激增，这些数据集涵盖学术环境中的各个领域。然而，并非所有数据集都能直接用于 IURS 场景。事实上，用于 IURS 的数据集通常需要满足以下两个条件。首先，观测到的交互应带有具体的时间戳，否则问题就会简化为只考虑位置顺序，从而忽略时间跨度，无法体现时间效应。其次，每次交互标注的时间戳应表示交互实际发生的时间，即用户 u 观看电影 i的时间，而不是用户 u对电影点赞或点踩反馈的时间 i。尽管有些此类数据集已用于 IURS，但部分研究人员 [4] 质疑这些数据集实际上并不适用于 IURS 这一特定问题。

表 1 列出了公开可用的基准数据集及其统计信息，这些数据集常用于 IURS 相关文献。请注意，IURS 列表示上述数据集是否确实适用于 IURS 场景。例如，像 *MovieLens* 和 *Netflix* 这样的数据集显然不适用于 IURS，因为物品上标注的时间戳是用户产生反馈的时间，而不是用户实际观看电影的时间。在这种情况下，无法保证交互之间的时间顺序。相比之下，用于音乐兴趣推荐的音乐艺术家数据集 *Last.fm* 由于具有实时反馈的特性，因此更为合适。

6 未来研究方向

推荐系统的增量更新是多种在线服务中的关键任务之一，其目标是随着新数据流入而即时更新模型，并基于新到达的数据，针对当前用户活动生成令人满意的推荐结果。尽管本综述回顾了以往关于 IURS 的研究，但仍存在若干重要的开放性研究问题，其中一些将在下文中提及。

6.1 动态系统建模

大型电子商务平台的用户-物品交互网络本质上是动态的，其中相互关联的物品会随着时间推移呈现复杂的行为。这类系统通常具有以下特征：

(1) 节点本身会自我演化。例如，2017年，iPhone 8 代表最新款设备；而到了2019年，它则代表价格实惠的设备。(2) 节点会受到图中邻居的影响。(3) 边会受到源节点和目标节点特征的影响。(4) 边本身也会自我演化。例如，由于某些季节性因素（如假期、节日），两个物品之间的交互会自然发生变化，这种变化不一定与物品特征的变化有关。

学习能够反映节点在网络中所起作用的优质表示，是分析和挖掘动态系统中用户兴趣变化的有力工具。随着近期图深度学习取得显著进展，研究者重新关注由深度神经网络参数化的网络表示学习。 [73, 79, 116]然而，大多数网络表示学习方法都是针对静态场景设计的，无法适应快速变化的动态网络。为适应动态网络环境，研究者开发了许多动态网络表示学习方法[77, 84, 97, 103]。这些方法大多将动态网络视为不同时间点快照的集合。每个快照由一段时间内的节点和边构成。在表示学习中，这些方法根据时间 t 的快照信息及其之前的快照，为该时间 t 生成嵌入表示。这种方法只能粗略估计不同快照间潜在空间中的节点嵌入变化 t，却不可避免地忽略节点状态变化的连续性。

[32] Fan 等人将用户/物品嵌入与生成的连续时间嵌入拼接起来，以统一序列模式和时序协同信号。Choi 等人 [21] 首次将神经常微分方程（NODEs）应用于动态推荐系统建模。Guo 和 Zhang 等人[40] 进一步将 NODEs 的思想拓展到连续时间时序会话图。他们相较于基线方法取得了显著的性能提升，这为以完全连续的方式建模系统动态带来了启发。

6.2 增量学习

研究人员提出了两种主流的持续学习方法，以克服推荐系统增量更新中特有的遗忘问题，即基于样本的方法和基于模型的方法。尽管这两种方法都具有一定的有效性，但仍存在一些主要局限。对于基于样本的方法，尽管其旨在选择并保留最具代表性的样本，但这些选定的样本仍不足以全面反映数据分布。为解决增量更新问题，研究人员提出了基于模型的方法，该方法考虑了来自先前模型和当前模型的知识迁移。然而，当前基于模型的方法存在一个主要局限：其中大多数很少考虑过去的知识是否适用于当前任务。因此，与其直接使用所有过去的代表性知识重新训练模型，未来的工作应当判断过去的知识是否适用于当前任务。 [33] 指出，减少重叠表示能够减少导致灾难性遗忘的干扰。神经生物学领域的近期研究 [104] 也支持这一观点：一个激活神经元会通过“侧向抑制”抑制其较弱邻居的活动。借助这一机制，大脑能够形成强大的去相关且紧凑的表示，从而将不同输入模式带来的干扰降至最低[6]。因此，学习去相关表示或许有助于对抗灾难性遗忘。此外，借助此前提出的跨任务对齐梯度的思想 [61], ，可以建立与元学习器的非平凡联系，使其实现相同目标：进一步使无关的已学习表示去相关，同时促进相关已学习表示之间的知识迁移。未来，这类方法还可以扩展到 IURS。

6.3 检索模块

许多现有方法通过为每个用户对所有物品进行评分来预测用户-物品交互 [9, 23, 108, 116], ，其时间复杂度为线性。这使得此类方法无法扩展到包含数百万次交互的数据集。Kumar 等人 [53] 生成预测物品的嵌入，而不是交互分布。他们推荐嵌入空间中距离预测物品最近的物品。结合局部敏感哈希方法，他们成功将检索时间从线性时间复杂度降低到常数时间。Zhou 等人 [114] 提出将先前批次中已计算的物品表示存储到队列中。然后，他们采用该队列作为负样本的有效采样器。得益于计算复用，这种类队列设计在深度候选生成中实现了显著的效率提升。该方法已部署上线，在淘宝每天服务数十亿次页面浏览量。然而，如何高效检索推荐列表仍然是一个尚未解决的问题。

6.4 IURS 的评估

以往与 IURS 相关的工作仅考虑使用精确率或平均倒数排名等指标评估推荐准确率。然而，物品多样性、推荐公平性等其他问题在在线平台上也至关重要。探索推荐系统评估流程中的偏差问题也很有意义 [101**?**]。例如，

推荐模型会产生偏向热门物品的偏差[67]。热门物品更有可能获得更多反馈，而这些反馈又可能影响未来推荐，并助长具有误导性的马太效应。在这种情况下，模型会偏向推荐热门物品，而无法推荐相关的长尾物品（不太热门或出现频率较低的物品）。如何处理所谓的流行度偏差问题，在推荐系统增量更新过程中具有很强的现实意义。

6.5 与推荐系统其他领域的联系

另一个研究方向是探索与推荐系统其他领域的联系。例如，值得尝试整合丰富的侧信息，如混合推荐系统 [5]，跨域推荐系统 [14] 和上下文感知推荐系统 [2, 31]。此外，还可以跟进持续学习领域的最新研究，重点解决从头训练问题。例如， [66] 首次将持续学习引入基于会话的推荐系统，这是弥合批量推荐系统与 IURS 之间差距的重要一步。未来的研究也可以考虑设计这类方法。

7 结论

本文系统综述了神经推荐系统的增量更新。我们介绍了增量更新推荐系统（IURS）的关键概念，并对这一任务进行了形式化定义。随后，我们阐述了 IURS 相较于传统批量更新推荐系统（BURS）所面临的挑战。此外，我们根据现有文献所应对的具体挑战，对其进行了整理和分类。之后，我们详细阐述了 IURS 中的评估问题。我们还指出了一系列开放问题和有前景的未来拓展方向。神经推荐系统的增量更新在实际工业应用中具有重要价值。我们希望本综述能帮助读者全面了解这一领域的关键方面，明确最显著的进展，并激发这一具有实践意义的研究领域中更多的研究。

参考文献

[1] Gediminas Adomavicius 和 Alexander Tuzhilin。2005。迈向新一代推荐系统：现状综述与可能的扩展。《*IEEE* 知识与数据工程汇刊》17，6（2005），734–749。[2] Gediminas Adomavicius 和 Alexander Tuzhilin。2011。上下文感知推荐系统。载于推荐系统手册。Springer，217–253。[3] Marie Al-Ghossein、Talel Abdessalem 和 Anthony Barré。2018。用于在线推荐的动态局部模型。载于*2018*年 *The Web Conference* 伴随论文集。1419–1423。[4] Marie Al-Ghossein、Talel Abdessalem 和 Anthony BARRÉ。2021。基于流的推荐系统综述。*ACM*计算调查（*CSUR*）54，5（2021），1–36。[5] Marie Al-Ghossein、Pierre-Alexandre Murena、Talel Abdessalem、Anthony Barré 和 Antoine Cornuéjols。2018。用于在线推荐的自适应协同主题建模。载于第*12*届 *ACM*推荐系统大会论文集。338–346。[6] Rahaf Aljundi、Marcus Rohrbach 和 Tinne Tuytelaars。2019。无私序贯学习。*ICLR 2019*论文集（2019）。[7] áOscar Celma。2010。数字音乐领域的音乐推荐与发现：长尾、长败与长播。Springer。[8] Ting Bai、Lixin Zou、Wayne Xin Zhao、Pan Du、Weidong Liu、Jian-Yun Nie 和 Ji-Rong Wen。2019。CTrec：用于连续时间推荐的长短期需求演化模型。载于第*42*届国际 *ACM SIGIR* 信息检索研究与发展会议论文集。675–684。[9] Alex Beutel、Paul Covington、Sagar Jain、Can Xu、Jia Li、Vince Gatto 和 Ed H Chi。2018。潜在交叉：在循环推荐系统中利用上下文。载于第*11*届 *ACM* 国际网络搜索与数据挖掘会议论文集。46–54。[10] Magdalena Biesialska、Katarzyna Biesialska 和 Marta R Costa-jussà。2020。自然语言处理中的持续终身学习：综述。*arXiv*预印本 *arXiv:2012.09823*（2020）。[11] Lukas Brozovsky 和 Vaclav Petricek。2007。在线约会服务的推荐系统。*arXiv*预印本 *cs/0703042*（2007）。[12] Peter Brusilovsky、Iván Cantador、Yehuda Koren、Tsvi Kuflik 和 Markus Weimer。2010。推荐系统中信息异质性与融合研讨会（HetRec 2010）。载于第四届 *ACM*推荐系统大会论文集。375–376。

[13] Pedro G Campos、Fernando Díez 和 Iván Cantador。2014。时间感知推荐系统：现有评估协议的全面综述与分析。《用户建模与用户适应交互》24，1（2014），67–119。[14] Iván Cantador、Ignacio Fernández-Tobías、Shlomo Berkovsky 和 Paolo Cremonesi。2015。跨域推荐系统。载于推荐系统手册。Springer，919–959。[15] Shiyu Chang、Yang Zhang、Jiliang Tang、Dawei Yin、Yi Chang、Mark A Hasegawa-Johnson 和 Thomas S Huang。2017。流式推荐系统。载于第26届万维网国际会议论文集。381–389。[16] Chen Chen、Hongzhi Yin、Junjie Yao 和 Bin Cui。2013。Terec：基于推文流的时间推荐系统。《VLDB基金会论文集》6，12（2013），1254–1257。[17] Tianwen Chen 和 Raymond Chi-Wing Wong。2020。处理基于会话的推荐中图神经网络的信息丢失。载于第26届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集。1172–1180。[18] Zeyuan Chen、Wei Zhang、Junchi Yan、Gang Wang 和 Jianyong Wang。2021。在按时间切片的用户-物品交互图上学习用于序列推荐的双重动态表示。载于第30届 ACM 国际信息与知识管理会议论文集。231–240。[19] Heng-Tze Cheng、Levent Koc、Jeremiah Harmsen、Tal Shaked、Tushar Chandra、Hrishi Aradhye、Glen Anderson、Greg Corrado、Wei Chai、Mustafa Ispir 等。2016。用于推荐系统的宽深学习。载于第1届推荐系统深度学习研讨会论文集。7–10。[20] Eunjoon Cho、Seth A Myers 和 Jure Leskovec。2011。友谊与移动：基于位置的社交网络中的用户移动。载于第17届 ACM SIGKDD知识发现与数据挖掘国际会议论文集。1082–1090。[21] Jeongwhan Choi、Jinsung Jeon 和 Noseong Park。2021。LT-OCF：可学习时间的基于ODE协同过滤。载于第30届 ACM 国际信息与知识管理会议论文集。251–260。[22] Paul Covington、Jay Adams 和 Emre Sargin。2016。用于 YouTube 推荐的深度神经网络。载于第10届 ACM推荐系统大会论文集。191–198。[23] Hanjun Dai、Yichen Wang、Rakshit Trivedi 和 Le Song。2016。深度协同进化网络：嵌入用户和物品特征以进行推荐。arXiv预印本 arXiv:1609.03675（2016）。[24] James Davidson、Benjamin Liebald、Junning Liu、Palash Nandy、Taylor Van Vleet、Ullas Gargi、Sujoy Gupta、Yu He、Mike Lambert、Blake Livingston 等。2010。YouTube视频推荐系统。载于第四届 ACM推荐系统大会论文集。293–296。[25] Gabriel de Souza Pereira Moreira、Felipe Ferreira 和 Adilson Marques da Cunha。2018。使用深度神经网络的基于新闻会话的推荐。载于第3届推荐系统深度学习研讨会论文集。15–23。[26] Matthias Delange、Rahaf Aljundi、Marc Masana、Sarah Parisot、Xu Jia、Ales Leonardis、Greg Slabaugh 和 Tinne Tuytelaars。2021。持续学习综述：在分类任务中抵御遗忘。《IEEE模式分析与机器智能汇刊》（2021）。[27] M Benjamin Dias、Dominique Locher、Ming Li、Wael El-Deredy 和 Paulo JG Lisboa。2008。个性化推荐系统对电子商务的商业价值：案例研究。载于2008年 ACM推荐系统大会论文集。291–294。[28] Ernesto Diaz-Aviles、Lucas Drumond、Zeno Gantner、Lars Schmidt-Thieme 和 Wolfgang Nejdl。2012。此时此刻发生了什么……让我感兴趣？Twitter上的在线主题发现与推荐。载于第21届 ACM 国际信息与知识管理会议论文集。1592–1596。[29] Ernesto Diaz-Aviles、Lucas Drumond、Lars Schmidt-Thieme 和 Wolfgang Nejdl。2012。社交流中的实时Top-N推荐。载于第六届 ACM推荐系统大会论文集。59–66。[30] Tracy Dong。2017。Flixster-dataset.zip。https://doi.org/10.6084/m9.figshare.5677741.v1[31] Zhengxiao Du、Xiaowei Wang、Hongxia Yang、Jingren Zhou 和 Jie Tang。2019。用于在线推荐的序列场景特定元学习器。载于第25届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集。2895–2904。[32] Ziwei Fan、Zhiwei Liu、Jiawei Zhang、Yun Xiong、Lei Zheng 和 Philip S Yu。2021。采用时间图协同变换器的连续时间序列推荐。载于第30届 ACM 国际信息与知识管理会议论文集。433–442。[33] Robert M French。1999。联结主义网络中的灾难性遗忘。《认知科学趋势》3，4（1999），128–135。[34] Erzsébet Frigó、Róbert Pálovics、Domokos Kelen、Levente Kocsis 和 András Benczúr。2017。非平稳环境中的在线排序预测。（2017）。[35] P Moreira Gabriel De Souza、Dietmar Jannach 和 Adilson Marques Da Cunha。2019。结合循环神经网络的上下文混合基于会话的新闻推荐。《IEEE Access》7（2019），169185–169203。[36] João Gama、Jesús Aguilar-Ruiz 和 Ralf Klinkenberg。2008。从数据流中发现知识。《智能数据分析》12（2008年5月），251–252。https://doi.org/10.3233/IDA-2008-12301[37] Joao Gama、Raquel Sebastiao 和 Pedro Pereira Rodrigues。2013。论流学习算法的评估。《机器学习》90，3（2013），317–346。[38] Carlos A Gomez-Uribe 和 Neil Hunt。2015。Netflix推荐系统：算法、商业价值与创新。《ACM管理信息系统汇刊（TMIS）》6，4（2015），1–19。[39] Asela Gunawardana 和 Guy Shani。2015。评估推荐系统。载于推荐系统手册。Springer，265–308。[40] Jiayan Guo、Peiyan Zhang、Chaozhuo Li、Xing Xie、Yan Zhang 和 Sunghun Kim。2022。通过图嵌套GRU ODE进行进化偏好学习，以实现基于会话的推荐。载于第31届 ACM 国际信息与知识管理会议论文集。

624–634.[41] Lei Guo、Hongzhi Yin、Qinyong Wang、Tong Chen、Alexander Zhou 和 Nguyen Quoc Viet Hung。2019年。流式基于会话的推荐。载于第25届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集。1569–1577。[42] Priyanka Gupta、Diksha Garg、Pankaj Malhotra、Lovekesh Vig 和 Gautam Shroff。2019年。NISER：用于处理流行度偏差的归一化物品与会话表示。arXiv 预印本 arXiv:1909.04276（2019年）。[43] F Maxwell Harper 和 Joseph A Konstan。2015年。MovieLens数据集：历史与背景。《ACM 交互式智能系统汇刊》（TIIS）5卷4期（2015年），1–19。[44] Xiangnan He、Kuan Deng、Xiang Wang、Yan Li、Yongdong Zhang 和 Meng Wang。2020年。LightGCN：简化并增强用于推荐的图卷积网络。载于第43届国际 ACM SIGIR 信息检索研究与发展会议论文集。639–648。[45] Zhankui He、Handong Zhao、Zhe Lin、Zhaowen Wang、Ajinkya Kale 和 Julian Mcauley。2021年。Locker：局部约束自注意力序列推荐。载于第30届 ACM 国际信息与知识管理会议论文集。3088–3092。[46] Balázs Hidasi 和 Alexandros Karatzoglou。2018年。具有Top-k增益的循环神经网络，用于基于会话的推荐。《第27届 ACM 国际信息与知识管理会议论文集》（2018年）。[47] Balázs Hidasi、Alexandros Karatzoglou、L. Baltrunas 和 D. Tikk。2016年。基于循环神经网络的会话推荐。CoRR abs/1511.06939（2016年）。[48] Geoffrey Hinton、Oriol Vinyals 和 Jeff Dean。2015年。提炼神经网络中的知识。arXiv 预印本 arXiv:1503.02531（2015年）。[49]]joachimsrecommendations Thorsten Joachims 和 CORNELL EDU。[无日期。]。将推荐视为处理：去偏学习与评估。（[无日期。]）。[50] Michael Jugovac、Dietmar Jannach 和 Mozhgan Karimi。2018年。StreamingRec：用于基准测试基于流的新闻推荐器的框架。载于第12届 ACM 推荐系统大会论文集。269–273。[51] Benjamin Kille、Frank Hopfgartner、Torben Brodt 和 Tobias Heintz。2013年。plista数据集。载于2013年国际新闻推荐系统研讨会与挑战赛论文集。16–23。[52] James Kirkpatrick、Razvan Pascanu、Neil Rabinowitz、Joel Veness、Guillaume Desjardins、Andrei A Rusu、Kieran Milan、John Quan、Tiago Ramalho、Agnieszka Grabska-Barwinska 等。2017年。克服神经网络中的灾难性遗忘。《美国国家科学院院刊》114卷13期（2017年），3521–3526。[53] Srijan Kumar、Xikun Zhang 和 Jure Leskovec。2019年。预测时序交互网络中的动态嵌入轨迹。载于第25届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集。1269–1278。[54] Patrick J Laub、Thomas Taimre 和 Philip K Pollett。2015年。霍克斯过程。arXiv 预印本 arXiv:1507.02822（2015年）。[55] J. Li、Pengjie Ren、Zhumin Chen、Z. Ren、Tao Lian 和 J. Ma。2017年。神经注意力会话推荐。《2017年 ACM 信息与知识管理会议论文集》（2017年）。[56] Jiacheng Li、Yujie Wang 和 Julian McAuley。2020年。用于序列推荐的时间间隔感知自注意力。载于第13届国际网络搜索与数据挖掘会议论文集。322–330。[57] Yuqi Li、Weizheng Chen 和 Hongfei Yan。2017年。学习基于图的嵌入，用于时间感知产品推荐。载于2017年 ACM 信息与知识管理会议论文集。2163–2166。[58] Qiao Liu、Y. Zeng、Refuoe Mokhosi 和 H. Zhang。2018年。STAMP：用于基于会话的推荐的短期注意力/记忆优先模型。《第24届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集》（2018年）。[59] Xin Liu 和 Karl Aberer。2014年。迈向动态Top-n推荐框架。载于第8届 ACM 推荐系统大会论文集。217–224。[60] Andreas Lommatzsch 和 Sahin Albayrak。2015年。面向用户-物品流的实时推荐。载于第30届 ACM 应用计算年会研讨会论文集。1039–1046。[61] David Lopez-Paz 和 Marc'Aurelio Ranzato。2017年。用于持续学习的梯度情景记忆。《神经信息处理系统进展》30卷（2017年），6467–6476。[62] Pawel Matuszyk 和 Myra Spiliopoulou。2014年。推荐系统中增量矩阵分解的选择性遗忘。载于国际发现科学会议。Springer，204–215。[63] Pawel Matuszyk 和 Myra Spiliopoulou。2017 年。用于推荐系统的基于流的半监督学习。《机器学习》 106 卷 6 期（ 2017 年），771–798。[64] Julian McAuley 和 Jure Leskovec。2013年。隐因子与隐主题：利用评论文本理解评分维度。载于第7届 ACM 推荐系统大会论文集。165–172。[65] Fei Mi 和 Boi Faltings。2020年。用于增量式会话推荐的记忆增强神经网络模型。arXiv 预印本 arXiv:2005.01573（2020年）。[66] Fei Mi、Xiaoyu Lin 和 Boi Faltings。2020年。ADER：面向基于会话的推荐持续学习的自适应蒸馏示例回放。载于第十四届 ACM 推荐系统大会。408–413。[67] Marco Morik、Ashudeep Singh、Jessica Hong 和 Thorsten Joachims。2020年。动态学习排序中的公平性与偏差控制。载于第43届国际 ACM SIGIR 信息检索研究与发展会议论文集。429–438。

[68] Shumpei Okura、Yukihiro Tagami、Shingo Ono 和 Akira Tajima。2017年。面向数百万用户的基于嵌入的新闻推荐。载于第23届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集。1933–1942。[69] Cosimo Palmisano 、 Alexander Tuzhilin 和 Michele Gorgoglione。2008年。利用上下文改进个性化应用中的客户预测建模。《IEEE 知识与数据工程汇刊》20卷11期（2008年），1535–1549。[70] Róbert Pálovics、András A Benczúr、Levente Kocsis、Tamás Kiss 和 ErzsébetFrigó。2014年。在线推荐中的时间影响利用。载于第8届 ACM 推荐系统大会论文集。273–280。[71] Z. Pan、Fei Cai、Wanyu Chen、Honghui Chen 和 M. Rijke。2020年。用于基于会话推荐的星型图神经网络。《第29届 ACM 国际信息与知识管理会议论文集》（2020年）。[72] Danni Peng、Sinno Jialin Pan、Jie Zhang 和 Anxiang Zeng。2021年。学习用于增量更新推荐系统的自适应元模型生成器。载于第十五届 ACM 推荐系统大会。411–421。[73] Bryan Perozzi、Rami Al-Rfou 和 Steven Skiena。2014年。DeepWalk：社会表示的在线学习。载于第20届 ACM SIGKDD 知识发现与数据挖掘国际会议论文集（美国纽约州纽约市）（KDD '14）。美国纽约州纽约市，701–710。https://doi.org/10.1145/2623330.2623732[74] Ruihong Qiu、Hongzhi Yin、Zi Huang 和 Tong Chen。2020年。GAG：用于流式基于会话推荐的全局属性图神经网络。载于第43届国际 ACM SIGIR 信息检索研究与发展会议论文集。669–678。[75] Massimo Quadrana、Paolo Cremonesi 和 Dietmar Jannach。2018年。序列感知推荐系统。《ACM计算调查》（CSUR）51卷4期（2018年），1–36。[76] Francesco Ricci、Lior Rokach 和 Bracha Shapira。2011年。推荐系统手册简介。载于《推荐系统手册》。Springer，1–35。[77] Aravind Sankar、Yanhong Wu、Liang Gou、Wei Zhang 和 Hao Yang。2020年。DySAT：通过自注意力网络在动态图上进行深度神经表示学习。载于 WSDM '20：第十三届 ACM 国际网络搜索与数据挖掘会议。519–527。https://doi.org/10.1145/3336191.3371845[78] Zaigham Faraz Siddiqui、Eleftherios Tiakas、Panagiotis Symeonidis、Myra Spiliopoulou 和 Yannis Manolopoulos。2014年。xstreams：为具有随时间演化偏好的用户推荐物品。载于第4届国际网络智能、挖掘与语义会议（WIMS14）论文集。1–12。[79] Uriel Singer、Ido Guy 和 Kira Radinsky。2019年。时间图上的节点嵌入。载于第二十八届人工智能国际联合会议论文集，IJCAI-19。4605–4612。https://doi.org/10.24963/ijcai.2019/640[80] Qingquan Song、Shiyu Chang 和 Xia Hu。2019年。耦合变分循环协同过滤。载于第25届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集。335–343。[81] R. Srivastava、Klaus Greff 和 J. Schmidhuber。2015年。高速公路网络。ArXiv abs/1505.00387（2015年）。[82] Karthik Subbian、Charu Aggarwal 和 Kshiteesh Hegde。2016年。流式数据推荐。载于第25届 ACM 国际信息与知识管理会议论文集。2185–2190。[83] Jiaxi Tang 和 Ke Wang。2018年。通过卷积序列嵌入实现个性化Top-N序列推荐。载于第十一届 ACM 国际网络搜索与数据挖掘会议论文集。565–573。[84] Rakshit Trivedi、Mehrdad Farajtabar、Prasenjeet Biswal 和 Hongyuan Zha。2019年。DyRep：动态图上的表示学习。载于国际学习表示会议。[85] Roberto Turrin、Massimo Quadrana、Andrea Condorelli、Roberto Pagano 和 Paolo Cremonesi。2015年。30Music听歌与播放列表数据集。。载于 RecSys Posters。[86] João Vinagre、Alípio Mário Jorge 和 João Gama。2015年。结合基于近期性的负反馈进行协同过滤。载于第30届 ACM 应用计算年会研讨会论文集。963–965。[87] João Vinagre、Alípio Mário Jorge 和 João Gama。2015年。流式环境下推荐系统的评估。arXiv 预印本 arXiv:1504.08175（2015年）。[88] Joao Vinagre、Alípio Mário Jorge 和 Joao Gama。2018年。推荐系统的在线装袋。《专家系统》35卷4期（2018年），e12303。[89] Jialei Wang、Steven CH Hoi、Peilin Zhao 和 Zhi-Yong Liu。2013年。面向即时推荐系统的在线多任务协同过滤。载于第7届 ACM 推荐系统大会论文集。237–244。[90] Jinshan Wang、Qianfang Xu、Jiahuan Lei、Chaoqun Lin 和 Bo Xiao。2020年。PA-GGAN：采用位置感知门控图注意力网络的基于会话推荐。2020年 IEEE多媒体与博览国际会议（ICME）（2020年），1–6。[91] Pengfei Wang、Jiafeng Guo、Yanyan Lan、Jun Xu、Shengxian Wan 和 Xueqi Cheng。2015年。为下一篮子推荐学习层次表示模型。载于第38届国际 ACM SIGIR 信息检索研究与发展会议论文集。403–412。[92] Qinyong Wang、Hongzhi Yin、Zhiting Hu、Defu Lian、Hao Wang 和 Zi Huang。2018年。采用对抗训练的神经记忆流式推荐网络。载于第24届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集。2467–2475。[93] Weiqing Wang、Hongzhi Yin、Zi Huang、Qinyong Wang、Xingzhong Du 和 Quoc Viet Hung Nguyen。2018年。基于流式排序的推荐系统。载于第41届国际 ACM SIGIR 信息检索研究与发展会议。525–534。[94] Yichao Wang、Huifeng Guo、Ruiming Tang、Zhirong Liu 和 Xiuqiang He。2020年。一种训练深度点击率模型的实用增量方法。arXiv 预印本 arXiv:2009.02147（2020年）。

[95] Ziyang Wang、W. Wei、G. Cong、X. Li、Xian-Ling Mao 和 Minghui Qiu。2020年。用于会话推荐的全局上下文增强图神经网络。载于第 43 届国际 ACM SIGIR 信息检索研究与发展会议论文集（2020年）。[96] Chuhan Wu、Fangzhao Wu、Suyu Ge、Tao Qi、Yongfeng Huang 和 Xing Xie。2019年。基于多头自注意力的神经新闻推荐。载于 2019年自然语言处理实证方法会议和第 9 届自然语言处理国际联合会议（EMNLP-IJCNLP）论文集。6389–6394。[97] Junshuang Wu、Richong Zhang、Yongyi Mao、Hongyu Guo、Masoumeh Soflaei 和 Jinpeng Huai。2020年。用于实体链接的动态图卷积网络。载于 WWW '20：The Web Conference。1149–1159。https://doi.org/10.1145/3366423.3380192[98] S. Wu、Y. Tang、Yanqiao Zhu、L. Wang、X. Xie 和 T. Tan。2019年。基于会话的图神经网络推荐。载于 AAAI。[99] Yishi Xu、Yingxue Zhang、Wei Guo、Huifeng Guo、Ruiming Tang 和 Mark Coates。2020年。Graphsail：面向推荐系统的图结构感知增量学习。载于第 29 届 ACM 国际信息与知识管理会议论文集。2861–2868。[100] Jaewon Yang 和 Jure Leskovec。2011年。在线媒体时间变化的模式。载于第四届 ACM 国际网络搜索与数据挖掘会议论文集。177–186。[101] Longqi Yang、Yin Cui、Yuan Xuan、Chenyang Wang、Serge Belongie 和 Deborah Estrin。2018年。针对非随机缺失隐式反馈的无偏离线推荐评估。载于第 12 届 ACM推荐系统大会论文集。279–287。[102] Feng Yu、Yanqiao Zhu、Qiang Liu、S. Wu、L. Wang 和 Tieniu Tan。2020年。TAGNN：用于基于会话的推荐的目标感知图神经网络。载于第 43 届国际 ACM SIGIR 信息检索研究与发展会议论文集（2020年）。[103] Wenchao Yu、Wei Cheng、Charu C. Aggarwal、Kai Zhang、Haifeng Chen 和 Wei Wang。2018年。NetWalk：一种用于动态网络异常检测的灵活深度嵌入方法。载于第 24 届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集。2672–2681。https://doi.org/10.1145/3219819.3220024[104] Yuguo Yu、Michele Migliore、Michael L Hines 和 Gordon M Shepherd。2014年。平衡与不平衡的树突间兴奋与抑制所产生的稀疏编码和侧向抑制。《神经科学杂志》34，41（2014），13701–13713。[105] Fajie Yuan、Alexandros Karatzoglou、Ioannis Arapakis、Joemon M Jose 和 Xiangnan He。2019年。一种用于下一项目推荐的简单卷积生成网络。载于第十二届 ACM 国际网络搜索与数据挖掘会议论文集。582–590。[106] Hui Zhang、Xu Chen 和 Shuai Ma。2019年。基于分层注意力网络的动态新闻推荐。载于 2019年 IEEE 国际数据挖掘会议（ICDM）。IEEE，1456–1461。[107] Peiyan Zhang、Jiayan Guo、Chaozhuo Li、Yueqi Xie、Jae Boum Kim、Yan Zhang、Xing Xie、Haohan Wang 和 Sunghun Kim。2023年。通过 Atten-Mixer 网络高效利用多层级用户意图进行基于会话的推荐。载于第十六届 ACM 国际网络搜索与数据挖掘会议论文集。168–176。[108] Shuai Zhang、Lina Yao、Aixin Sun 和 Yi Tay。2019年。基于深度学习的推荐系统：综述与新视角。《ACM计算调查》（CSUR）52，1（2019），1–38。[109] Yang Zhang、Fuli Feng、Chenxu Wang、Xiangnan He、Meng Wang、Yan Li 和 Yongdong Zhang。2020年。如何重新训练推荐系统？一种序列元学习方法。载于第 43 届国际 ACM SIGIR 信息检索研究与发展会议论文集。1479–1488。[110] Wayne Xin Zhao、Sui Li、Yulan He、Edward Y Chang、Ji-Rong Wen 和 Xiaoming Li。2015年。连接社交媒体与电子商务：利用微博信息进行冷启动商品推荐。《IEEE 知识与数据工程汇刊》28，5（2015），1147–1159。[111] Xin Wayne Zhao、Yanwei Guo、Yulan He、Han Jiang、Yuexin Wu 和 Xiaoming Li。2014年。我们知道你想买什么：一种用于微博商品推荐的基于人口统计信息的系统。载于第 20 届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集。1935–1944。[112] Yan Zhao、Shoujin Wang、Yan Wang 和 Hongwei Liu。2021年。用于流式推荐的分层时间感知采样自适应集成学习。《应用智能》51，6（2021），3121–3141。[113] Erheng Zhong、Wei Fan、Junwei Wang、Lei Xiao 和 Yong Li。2012年。Comsoc：在复合社交网络中自适应迁移用户行为。载于第 18 届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集。696–704。[114] Chang Zhou、Jianxin Ma、Jianwei Zhang、Jingren Zhou 和 Hongxia Yang。2021年。用于大规模推荐系统去偏候选生成的对比学习。载于第 27 届 ACM SIGKDD知识发现与数据挖掘国际会议论文集。3985–3995。[115] Huachi Zhou、Qiaoyu Tan、Xiao Huang、Kaixiong Zhou 和 Xiaoling Wang。2021年。用于基于会话的推荐的时间增强图神经网络。（2021年）。[116] L. Zhou、Y. Yang、X. Ren、F. Wu 和 Y. Zhuang。2018年。通过建模三元闭包过程进行动态网络嵌入。载于 AAAI。[117] Han Zhu、Daqing Chang、Ziru Xu、Pengye Zhang、Xiang Li、Jie He、Han Li、Jian Xu 和 Kun Gai。2019年。面向推荐系统的基于树的索引与深度模型联合优化。arXiv 预印本 arXiv:1902.07565（2019年）。[118] Han Zhu、Xiang Li、Pengye Zhang、Guozheng Li、Jie He、Han Li 和 Kun Gai。2018年。学习用于推荐系统的基于树的深度模型。载于第 24 届 ACM SIGKDD 国际知识发现与数据挖掘会议论文集。1079–1088。[119] Jingwei Zhuo、Ziru Xu、Wei Dai、Han Zhu、Han Li、Jian Xu 和 Kun Gai。2020年。束搜索下最优树模型的学习。载于国际机器学习会议。PMLR，11650–11659。