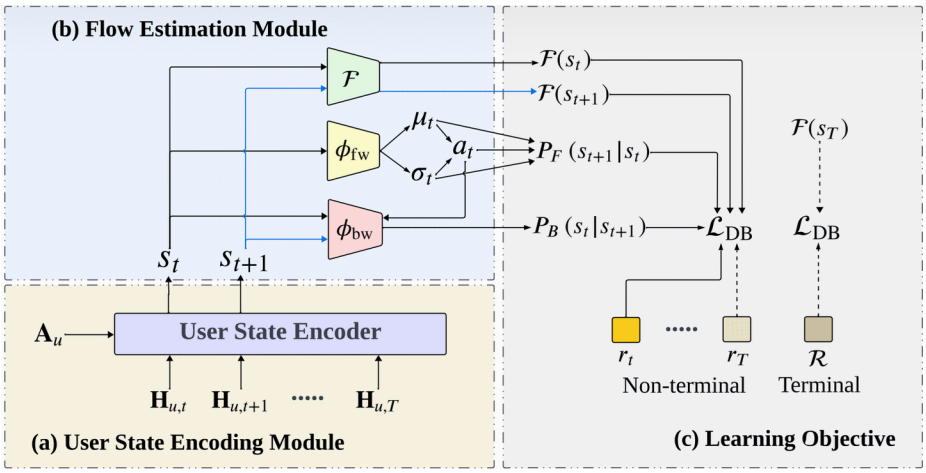


快手2024：GFN4Retention-通过生成流网络建模用户留存

SmartMindAI
专注搜索、广告、推荐、大模型和人工智能最新技术，欢迎关注我

Introduction

在信息泛滥的时代，推荐系统已经成为引导用户找到与其个人偏好相匹配内容的关键工具。用于评估这些系统的传统指标，如点击、喜欢和评分，捕捉用户对每个推荐内容的即时反馈，并将其作为优化推荐系统的指导目标。尽管这些指标非常有效，但它们本质上是在估计用户对内容的即时反馈，而无法提供用户长期参与度的全面评估。例如，当系统发现具有吸引人特点的内容（例如，具有上瘾内容的内容）能最大化点击率时，它可能会决定持续推荐这些内容。然而，这些特点可能在初期吸引用户，但很快就会失去用户的兴趣。这种差异表明，用户对内容的即时兴趣与系统的可持续兴趣之间存在差距。

为了解决这一问题，采用了长期指标来提供用户整体满意度的更深层次洞察。一个典型的例子是用户留存信号，描述了用户返回应用的行为。这个指标对于许多在线服务来说是最重要的性能评估之一，因为它与关键业务指标，即每日活跃用户（DAU）密切相关。 在实践中，建模和优化用户留存是一个具有挑战性的任务。具体来说，留存行为发生在用户离开当前会话并返回到下一个会话的开始时。它与用户在先前交互中的任何单个推荐步骤都没有明确关系。此外，用户在两个连续会话之间的活动对于服务来说是不可见的，这增加了额外的不确定性。

为应对这些挑战，使用强化学习（RL）作为代理来优化整个用户交互序列的累积奖励，已经显示出有希望的结果。直观地，用户返回平台是因为系统整体给人的印象足够积极和吸引人，这可以通过会话中与正面反馈相关的奖励之和来部分衡量。基于RL的推荐解决方案通过将用户交互序列建模为马尔可夫决策过程（MDP）来解决累积奖励的优化问题，并学习一个策略，考虑每个推荐行动的长期影响。这使得它们能够在用户交互的每个点动态地调整推荐，适应不断变化的偏好，并优化整个会话的累积奖励。然而，会话级累积奖励与用户留存之间的关系仍然不明确，因此本文进一步展示了基于强化学习的方法可以直接将跨会话留存信号整合到长期价值估计中。这种方法虽然有效，但并未设计用于探究每个交互对留存信号的影响（称为"留存归因"），并且它也间接通过累积即时奖励作为代理来优化留存。此外，所有基于强化学习的解决方案都可能面临探索与利用的权衡，这限制了它们在不稳定的指标上的性能。这种不稳定性在用户留存动态复杂、不确定且快速演变的场景中尤为明显。总的来说，我们希望有一个稳定的探索性解决方案，同时优化用户留存和即时奖励。

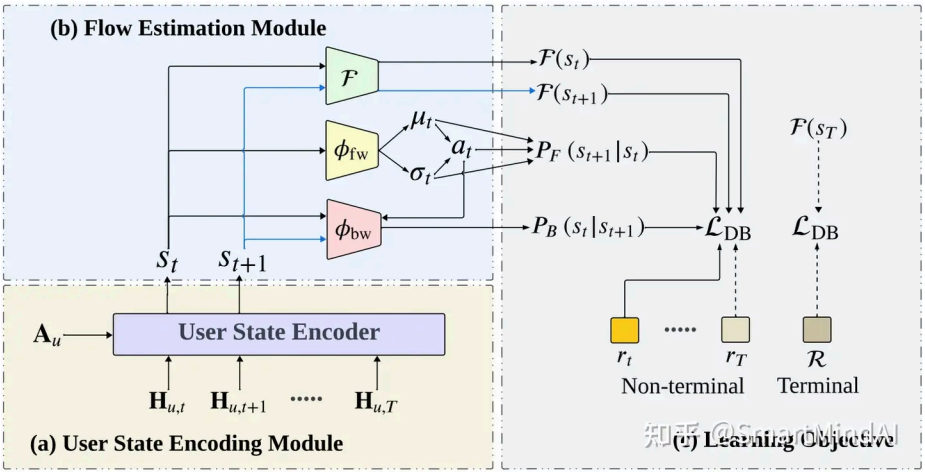
受到最近生成式网络（GFNs）发展的影响，我们提出了一种方法**GFN4Retention**，将会话级推荐视为生成任务，其中留存信号直接由轨迹生成概率建模。类似于GFN的一般形式，生成过程最终将构建一个用户会话（轨迹），只有在过程结束时，目标留存奖励才能匹配。具体来说，每个推荐步骤被视为对下一个用户状态的条件前向概率流，每个用户状态都与一个流估计器相关联，该估计器代表达到该状态的生成概率。在优化过程中，会话结束的终端状态直接将生成概率与留存奖励匹配。对于非终端状态，我们使用包含额外的反向概率流的学习目标来进行反向传播，以将留存奖励传递到序列中的每一步。这将隐式地建模每个推荐动作的留存归因。

知乎

能难以跟踪每个推荐步骤中用户参与的微妙变化，因为它最初假设不存在中间奖励。作为对策，我们推导了一个综合奖励系统，可以在每个推荐步骤中平衡即时奖励和留存归因。首先，将会话级生成概率与匹配留存奖励和即时反馈的无折现累积奖励相匹配。其次，推荐空间非常庞大，实际中通过连续向量空间中的一个点来生成和表示推荐列表。我们展示了在连续空间中，流匹配性质仍然成立，并说明了如何根据具体情况设计每个流估计组件。

Preliminaries

Generative Flow Networks



近年来，生成流网络（GFNs）已成为生成建模领域的一股创新力量，引入了一种在复杂分布中学习和采样的方法论。GFNs设计的核心在于其能够发展出一个生成策略，该策略引导状态空间*中的概率流动朝向终端状态。

具体来说，GFNs的通用公式考虑了一条生成轨迹

$$\mathbf{S} = \{s_1 \to s_2 \to \cdots \to s_T\}$$

并假设了一个向前的概率策略$P_F(s_{t+1}|s_t)$，该策略在生成过程中决定了每一步的动作，其中s_t表示第t步的状态T是轨迹的终端步骤。

GFNs的目标是通过最终的观察奖励，使每条轨迹的生成概率与之对齐： $P(\mathbf{S}) \propto R(s_T)$

这是一种本质上是能量驱动的方法，因为该模型同时扮演生成器*和评估器的角色。

在优化过程中，GFN框架引入了流估计器$\mathcal{F}(s_t)$，用于评估通过状态s_t的可能性。GFN中的流学习过程精确调整，以与目标分布保持平衡，确保流入和流出流的总和达到均衡：

$$\mathcal{F}(s_t) P_F(s_{t+1} \mid s_t) \approx \mathcal{F}(s_{t+1}) P_B(s_t \mid s_{t+1})$$

其中，从s_t到s_{t+1}的前向概率是 $P_F(s_{t+1} \mid s_t)$

而相应的后向概率是 $P_B(s_t \mid s_{t+1})$ 用于建模给定结果状态s_{t+1}的源状态s_t的可能性。

关于GFNs的基础工作导致了一种针对此目标的优化变体的发展：详细的平衡（DB）损失，它最小化了每一步联合概率*$P(s_t, s_{t+1})$的前向视图和后向视图之间的差异。然后，目标在生成过程中遵循了最小化详细平衡（DB）损失的流匹配特性：

$$\min \mathcal{L}_{\text{DB}}(s_t, s_{t+1}) = \left(\log \frac{\mathcal{F}(s_t) P_F(s_{t+1}|s_t)}{\mathcal{F}(s_{t+1}) P_B(s_t|s_{t+1})}\right)^2$$

通过最小化数据库损耗，它力求实现一个高度指示目标分布的数据基表示。

Problem Definition

知乎

形式上，我们考虑用户集合$\mathcal{U}$和内容集合$\mathcal{C}$。对于每个会话，在任意时刻t，我们可能会收到用户$u \in \mathcal{U}$的推荐请求，该请求包含用户特征集$\mathbf{A}_u$，以及用户到目前为止的交互历史$\mathbf{H}_{u,t}$。

在推荐系统中，推荐请求提供了编码当前用户状态s_t所需的上下文信息。给定推荐请求和编码状态，推荐策略生成一个动作a_t，该动作对应于从$\mathcal{C}$中选择的一份内容列表。

然后，用户对这些内容给出行为反馈，包括点击、喜欢和评论等行为类型，这些反馈用于计算一个即时奖励r_t：

$$r_t = \sum_{b \in \mathcal{B}} \omega_b \cdot y_{t,b}$$

其中$y_{t,b}$表示用户在步骤t对行为b的反馈，而ω_b是行为b的权重。 在会话结束时（即在s_T），我们还会观察到用户留存奖励$\mathcal{R}$，这是本工作中核心指标的定义，即用户的回访频率。 我们将每个样本组织为元组[+]

$$(\mathbf{S}, a_1, \ldots, a_T, r_1, \ldots, r_T, \mathcal{R})$$

其中包含了状态（即用户请求）、动作、即时奖励以及会话的留存奖励。

并且我们在问题设置中设定了两个目标：

1）寻找一个有效的奖励设计：

$$R(\mathbf{S}) = f(r_1, \ldots, r_T, \mathcal{R})$$

它结合了留存奖励和累计即时奖励，有助于提升整体推荐性能；

2）学习一个推荐策略，探索并实现更好的联合奖励$R(\mathbf{S})$。

THE PROPOSED Framework

User State Encoding

在现实世界的应用中，网络平台的实现高度依赖于理解和应对复杂的用户动态以及平台的静态特性。在这种情况下，优化留存策略不仅需要识别用户行为的多样性，还需要快速适应这些行为模式的变化。实践表明，在收到用户$u \in \mathcal{U}$的推荐请求时，我们考虑了两种主要类型的输入，包括用户的特征集$\mathbf{A}_u$和交互历史$\mathbf{H}_{u,t}$。

为了更好地利用用户历史中的动态因素，并在内容之间找到影响模式，我们首先使用一个变换器处理历史$\mathbf{H}_{u,t}$，并将最后的输出嵌入视为历史编码。接着，我们从$\mathbf{A}_u$生成一个用户特征嵌入，并将其与历史编码相连接，然后通过神经网络生成一个嵌入$\mathbf{e}_u$，这是用户状态s_t的第一部分。

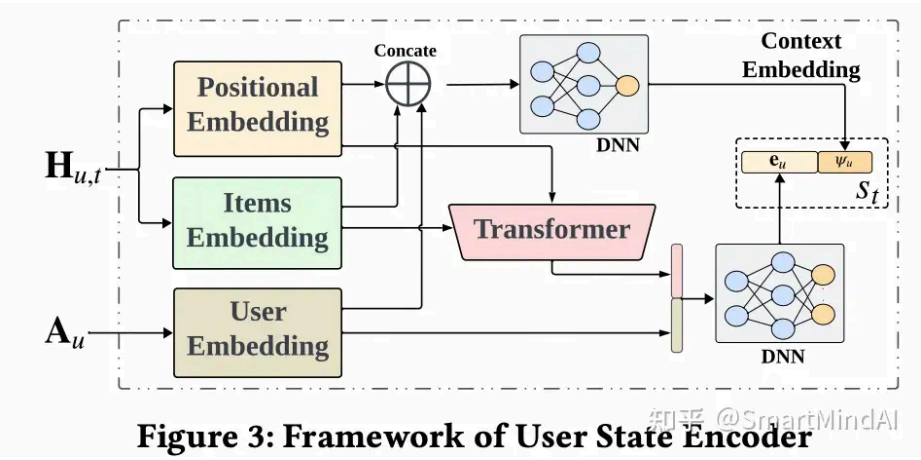


Figure 3: Framework of User State Encoder

实践表明，仅使用变换器编码用户历史可能会过度放大最近的历史信息，而忽视了特征级别的交互。因此，我们引入了一个基于深度神经网络[+]（DNN）的上下文检测模块，用于编码用户请求中

知乎

Recommendation Policy as Forward Flow

在推理阶段，通过编码的用户状态s_t，我们可以生成推荐的动作。在许多推荐系统中，服务需要为每个用户请求提供一系列内容列表，以满足用户频繁浏览行为的延迟需求。这意味着直接将庞大的内容集$\mathcal{C}$作为动作空间进行考虑是不实际的。相反，我们选择为每个动作a_t考虑一个向量空间，这个向量可以通过确定性的Top-K选择模块来表示输出内容列表。

在设计中，策略网络+（即前向流估计器）将首先输出高斯分布+的统计量 $\mu, \sigma = \phi_{\text{fw}}(s_t)$，然后通过$\mathcal{N}(\mu, \sigma)$采样动作向量作为$a_t$。

在训练阶段，推荐策略被视为前向流函数$P_F(s_{t+1}|s_t)$，假设输出动作a_t决定了下一个状态s_{t+1}。

Retention Flow Estimation

基于GFNs的一般设计，我们包含了状态流估计器$\mathcal{F}(s_t)$和一个反向流函数$P_B(s_t|s_{t+1})$，用于估计状态的后验概率+。

在会话层面的观点中，每个观察到的会话$\mathbf{S}$是由推荐策略生成的概率轨迹。根据P_F中的采样过程，状态s_t可能分裂成不同的未来状态，并且它可以由各种先前的状态达到。直观上，状态流估计器$\mathcal{F}(s_t)$表示状态s_t被达到的可能性。

反向函数 $P_B(s_t|s_{t+1}) = \phi_{\text{bw}}(s_t, a_t, s_{t+1})$

接受当前的状态动作和下一个状态作为输入，估计当前状态如何生成下一个状态的可能性。

为了确保属性$\mathcal{F}(s_t) \geq 0$和$P_B(\cdot) \geq 0$，我们使用了sigmoid激活函数+作为网络输出。 然后，我们可以将轨迹生成的可能性与观察到的留存奖励相匹配，并使用流匹配目标, 将这个会话结束时的留存信号反向传播到每个中间步骤：

$$\mathcal{L}_{\text{DB}} = \begin{cases} \left(\log \frac{\mathcal{F}_R(s_t) \cdot P_F(s_{t+1}|s_t)}{\mathcal{F}_R(s_{t+1}) P_B(s_t|s_{t+1})}\right)^2 & 1 \leq t \leq T-1 \\ \left(\log \frac{\mathcal{F}_R(s_t)}{\mathcal{R}}\right)^2 & t = T \end{cases}$$

在推荐策略ϕ_{fw}的影响下，前向函数$P_F(s_{t+1}|s_t)$依赖于推荐策略ϕ_{fw}。 这种设计确保了学习目标 $P(\mathbf{S}) \propto \mathcal{F}_R(s_T) \approx \mathcal{R}$

在学习过程中，生成可能最初是随机的，生成的留存奖励较低，但策略的探索效果会逐渐发现具有更高留存的样本，而在那些会话中的行动将获得更高的生成可能性。 最终，这种流量估计学习框架有助于生成策略提供更多的多样性，同时保持高质量。

Refined Detailed Balance Learning with Reward Integration

仅仅像前一节那样优化留存率+只能建模用户的长期偏好。相比之下，会话中每个中间步骤的即时奖励提供了有价值的细微信息，这些信息可能会被长期留存奖励忽略。例如，一个包含相关项目和不相关项目的会话仍然可能获得良好的留存奖励，因为只要有相关信息，用户可能会返回。然而，我们不应该将这两个项目视为相等，用户对每个项目的反馈可以帮助我们区分它们。直观地，我们认为留存奖励和即时奖励是用户偏好的互补视角。

Reward Design:

为了适应流量估算框架，确保详细平衡目标的正确性，我们提议将奖励通过乘积整合：
$R(\mathbf{S}) = \mathcal{R} \cdot e^{\alpha \cdot \sum_{t=1}^{T-1} r_t}$

其中α是参数，用于平衡用户即时奖励的重要性。

累计奖励$e^{\sum_{t=1}^{T-1} r_t}$是非折现的，并以指数形式呈现。这种设计与监督学习中的二元交叉熵+以及基于RL的解决方案（如TD3）相似，每一步都专注于学习对数尺度的策略输出，与即时奖励相关联。

知乎

Integrated Flow Matching

基于上述奖励整合，给定状态s_t对应的流量估计器被分解为两个对应的组成部分：

$$\mathcal{F}(s_t) = \mathcal{F}_R(s_t) \cdot (\mathcal{F}_I(s_t))^\alpha$$

其中$\mathcal{F}_R(s_t)$对应着留存奖励的流动$\mathcal{F}_I(s_t)$则对应着步骤t之前的累积即时奖励：

$$\mathcal{F}_I(s_t) = e^{\sum_{j=1}^{t-1} r_j}$$

不同于$\mathcal{F}_R$需要与GFN模块协同优化$\mathcal{F}_I$是一个非参数函数，仅依赖于观察到的即时回报：

对于结束状态，我们得到 $\mathcal{F}_R(s_T) = \mathcal{R}$ 和 $\mathcal{F}_I(s_t) = e^{\sum_{j=1}^{T-1} r_j}$

然后，基于流匹配的目标，我们可以将整合奖励反向传播到中间步骤：

$$\mathcal{F}(s_t) \cdot P_F\left(s_{t+1}|s_t\right) = \mathcal{F}(s_{t+1}) \cdot P_B\left(s_t|s_{t+1}\right)$$

将等式1、等式2和等式3结合在一起，通过操作我们可以得出以下简化等式：

$$\begin{gathered}
\mathcal{F}_R(s_t)(\mathcal{F}_I(s_t))^\alpha P_F\left(a_t \mid s_t\right) = \mathcal{F}_R(s_{t+1})(\mathcal{F}_I(s_{t+1}))^\alpha P_B\left(s_t|s_{t+1}\right) \\
(e^{\sum_{j=1}^{t-1} r_j})^\alpha \cdot \mathcal{F}_R(s_t) P_F\left(s_{t+1} \mid s_t\right) = (e^{\sum_{j=1}^{t} r_j})^\alpha \cdot \mathcal{F}_R(s_{t+1}) P_B\left(s_t|s_{t+1}\right) \\
\mathcal{F}_R(s_t) P_F\left(s_{t+}|s_t\right) = e^{\alpha r_t} \cdot \mathcal{F}_R(s_{t+1}) P_B\left(s_t|s_{t+1}\right)
\end{gathered}$$

该等式说明，在准确的模型预测下，增加的行动概率与即时奖励的提升或未来留存奖励的潜在增加相关。

Integrated Detail Balance Objective

根据等式中的流量匹配，我们得出对数尺度详细平衡学习目标如下：

$$\mathcal{L}_{\mathrm{DB}} = \begin{cases} \left(\log \mathcal{F}_R(s_t) + \log P_F\left(s_{t+1}|s_t\right) - \log \mathcal{F}_R(s_{t+1}) \right. \\ \quad \left. - \log P_B\left(s_t|s_{t+1}\right) - \alpha \cdot r_t\right)^2 & 1 \leq t \leq T-1 \\ \left(\log \mathcal{F}_R(s_t) - \log \mathcal{R}\right)^2 & t = T \end{cases}$$

在每个步骤的即时奖励r_t出现且仅在对应的步骤内数据库损失的步骤中出现，而终止状态在$t = T$时不观察到即时奖励，仅匹配留存流。系统地，我们为即时奖励设计的流$\mathcal{F}_I$是非参数化*的，并且可以通过一个简单的额外术语自然地集成到DB目标中。 相反，留存奖励需要学习$\mathcal{F}_R$，而流的反向传播与流匹配隐含地实现了"留存归因"。整体学习框架优化了集成目标，与$\mathcal{F}(s_T)$成正比，等于留存流$\mathcal{F}_R(s_T)$与即时奖励流$\mathcal{F}_I(s_T)$的α次幂。总的来说，我们将留存和即时奖励视为推荐政策用户感知的两个互补方面。在我们的解决方案中，即时奖励在每个步骤中提供直接指导，而留存流则以步骤归因为政策提供指导。 此外，向前概率$P_F(\cdot)$的值可能会在对数尺度上接近零，显著偏离流估计的有效区域。因此，我们将其作为超参数包括进来，并且相应的对数尺度估计变为$\log(P_F(\cdot) + \beta_F)$同样，我们也包括$\beta_B$来稳定反向函数的学习，并且$\beta_r$来减少奖励的方差。

Experiment

Dataset

我们使用了两个实际世界的数据集来执行我们的实验。

- Kuairand-Pure是一个非偏见的序列推荐数据集，其特点是随机视频展示。由其特征定义，其目的是提供一个无偏见的环境，用于评估和测试推荐算法。其网址为：kuairand.com/
- MovieLens-1M，在推荐系统领域广泛使用的一个基准，尽管规模更广泛，使得分布更稀疏。

Kuairand-Pure	27,285	7,551	1,436,609	0.70%
MovieLens-1M	6,400	3,706	1,000,209	4.22%

Overall Performance

为了评估我们提出的GFN4Retention模型的有效性，我们在这两个数据集上将其综合性能与五种基线模型进行了比较分析。详细结果如表所示。

Table 2: Overall Performance on two datasets for different models.

Dataset	Metric	Model					
		TD3	SAC	DIN	CEM	RLUR	GFN4Retention
Kuairend-Pure	Return Time	2.382	2.373	1.947	1.889	1.786	**1.496***
	Retention	0.151	0.150	0.154	0.156	0.159	**0.163***
	Click Rate	0.800	0.801	0.773	0.762	0.789	**0.805**
	Long View Rate	0.791	**0.795**	0.764	0.757	0.778	0.794
	Like Rate	0.852	0.857	0.812	0.804	0.831	**0.862**
ML-1M	Return Time	2.258	2.246	1.893	1.814	1.723	**1.479***
	Retention	0.141	0.142	0.153	0.158	0.160	**0.165***
	Click Rate	0.461	0.468	0.454	0.448	0.459	**0.473**
	Long View Rate	0.459	0.463	0.455	0.453	0.457	**0.464**
	Like Rate	0.568	0.571	0.541	0.524	0.561	**0.574**

通过这些观察，我们发现： 1. GFN4Retention模型在两个数据集上的整体表现优于五种基线模型。 2. 特别是，在返回日指标上，GFN4Retention模型展现出更优的趋势。 3. 这些结果证明了GFN4Retention模型在预测和优化返回日行为方面的潜力和优势。

- TD3模型在留存指标方面表现最弱。在两个数据集中的"返回时间"都较高，这表明用户会话之间的间隔更长，从而导致较低的留存率。这个模型在任何指标上都没有表现出色，可能是因为它对环境分布的变化适应性较差，而且其策略与特定用户行为模式的关联性较弱，导致了性能的不理想。
- 在各种基线模型中，RLUR模型在留存指标上表现突出。其设计巧妙地捕捉了用户留存动态，考虑到序列推荐任务中的固有偏见。尽管在优化即时用户反馈方面，其结果也十分出色，但训练过程中，RLUR模型的波动性较大，需要更多的迭代来达到收敛。
- SAC模型在优化实时用户反馈方面展现出卓越的基础性能。在所有指标下，它都具有竞争力，并在Kuairend-Pure数据集的长视野率方面处于领先地位。其通过平衡预期回报和策略熵的方法，从而使其能够有效地建模用户参与。
- 在多个重要指标下，我们的GFN4Retention模型超越了所有其他模型，甚至包括最佳基线模型。它实现了最低的返回时间，显示用户参与度更为频繁，同时在留存率和喜欢率方面获得了最高得分，且有显著的统计学改善。通过精细组织的方式，将即时反馈与最终留存信号相结合，GFN4Retention不仅提升了用户留存率⁺，还保持了即时用户反馈的质量。该模型的稳定性和可靠性还通过在所有基线模型中的最稳定训练曲线得到了进一步的验证。

总结而言，GFN4Retention模型通过有效平衡即时参与与用户留存，展现出卓越的性能。其在关键指标上取得的领先得分，以及相较于基线模型的显著改进。

原文《Modeling User Retention through Generative Flow Networks》

发布于 2024-08-16 17:40 · IP 属地北京

快手 推荐系统 用户留存