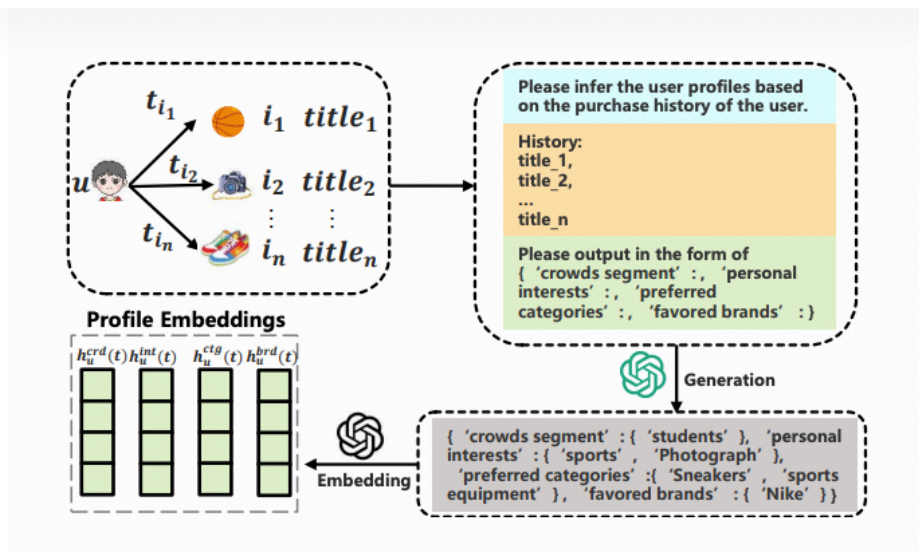




赞同 9

分享



2024阿里：DynLLM—时间动态推荐与LLM技术融合



SmartMindAI

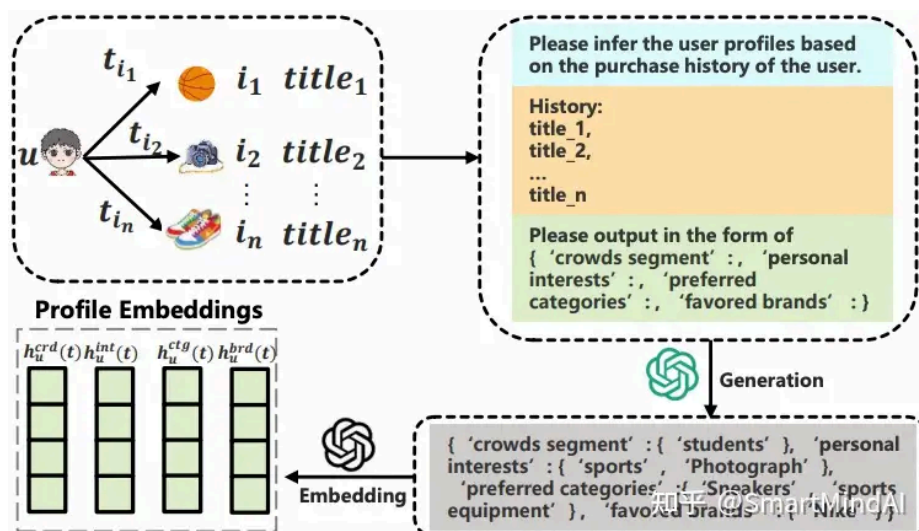
专注搜索、广告、推荐、大模型和人工智能最新技术，欢迎关注我

已关注

9 人赞同了该文章

Introduction

推荐系统⁺是处理信息过剩的利器，如电商、社交和视频平台的个性化推荐。CF是推荐的核心，尤其依赖GCN技术因能捕捉高阶关联而显著。然而，CF受限于数据稀疏性问题，LLMs的新颖特性为其提供了解决方案。我们的研究DynLLM通过利用LLMs创建详细用户画像，结合动态图⁺信息，突破了数据不全的限制，实现了更精准且理解用户行为变化的个性化推荐。



本文研究聚焦于如何利用LLMs的生成能力，辅以生成辅助信息⁺和元路径提示，以增强图基推荐，解决数据难题，揭示潜在模式。然而，目前LLMs还需与图数据处理能力相结合，以深化理解动态用户行为和兴趣，提供更个性化和实时的推荐。我们提出DynLLM，利用LLMs生成用户画像，包括群体特性和个人兴趣等，通过蒸馏注意力清理噪声，增强关键信息。这种方法结合动态图和时间信息，通过迭代更新用户和商品嵌入，既利用图结构又考虑时间演变，实现了对动态用户行为的深入理解和实时个性化推荐。

Preliminaries

CTDG是一种动态模型，它不仅包含静态图的连接关系，还能够捕捉到时间演变的动态交互行为。在这样的背景下，我们定义了基于CTDG的动态推荐。这种推荐系统不仅考虑了用户和商品的静态

Graph Recommendation

在**连续时间动态图推荐**中，我们使用连续时间动态图（Continuous Time Dynamic Graphs, CTDGs）模型，这是一种动态的图结构，用户和物品的关系不是静止的，而是随时间变化的。图由用户 $U = \{u_1, u_2, \dots, u_M\}$ 和物品 $I = \{i_1, i_2, \dots, i_N\}$ 组成，通过交互矩阵 Y 来记录每个时刻 t 用户与物品的互动情况 $Y_{ui}(t)$ 为0或1，表示交互状态。

动态推荐的目标是通过学习函数 $g_{(u,i)} : (u, i, t) \rightarrow s_t$

来预测用户在特定时间 t 对物品 i 的兴趣概率，其中 s_t 是一个实数值 g 需考虑用户行为的动态性和物品的演变。核心目标是找到一个能同时捕捉当下兴趣和预测未来潜在交互的推荐策略。动态图推荐致力于优化这个函数，以提供即时且个性化的推荐服务。

Continuous Time Dynamic Graph

在**连续时间动态图推荐**中，我们以CTDGs为基础，用户和物品关系随时间动态演化。具体为 G 由在时间 t 的用户 U_t 和物品 I_t 构成 U_t 和 I_t 包含 M 和 N 个用户和物品，分别随时间变化。**时间序列+集合 \mathcal{T}** 标记所有时间点 Y_t 记录每个时间点的交互事件。

动态推荐通过函数 $g_{(u,i,t)}$ 估计用户在 t 对 i 的兴趣 s_t 为实数。目标是优化该函数，确保推荐既体现当前兴趣，又能预测未来潜在互动，提供即时个性化推荐。

Dynamic Recommendation with CTDGs

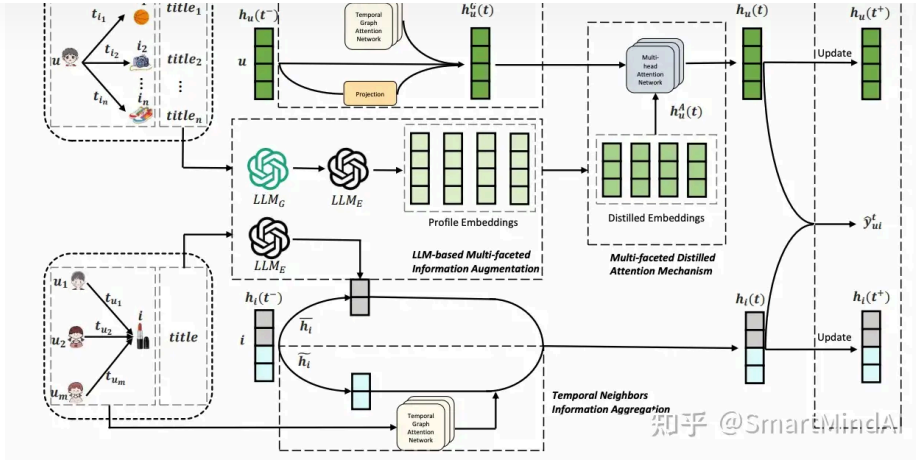
在**动态推荐问题表示**中，我们构建了一个模型 G ，包含用户集合 U 和物品集合 I ，以及在时间 t 的交互集合 Y_t 。 U 和 I 按时间变化 Y_t 记录用户对物品的交互事件。目标是通过学习函数 $g_{(u,i,t)}$ ，它将用户 t 对物品 i 的兴趣量化为 $s_t \in \mathbb{R}$ ，来预测用户未来可能的兴趣。

优化的目标是两方面：一是提升预测准确性，如衡量点击率或满意度；二是保证推荐的即时性，即减少推荐的延迟。这旨在提供个性化推荐服务，实时跟踪并适应用户行为变化。

在给定的连续时间动态图 G 中，动态推荐任务是设计函数 $f_{(u,i,t)}$ ，它根据用户在时间 t 对物品 i 的交互可能性($s_t \in \mathbb{R}$)进行预测。函数需要捕捉用户行为随时间的动态变化，生成针对每个用户和时间点的Top-K推荐列表。优化目标包括提高预测精度，比如通过衡量点击率或满意度来评估推荐的有效性。同时，推荐系统还需要确保推荐的即时性，即在接收到用户当前行为信息后尽快提供推荐。这样的目标是为了提供个性化且实时的服务，适应用户行为的瞬息万变。

Methods

该框架主要通过四个核心模块来利用语言模型技术提升动态图推荐：数据预处理模块、动态图建模模块、LLM集成模块、推荐生成模块。通过这些模块的协同工作，**DynLLM**不仅能更好地理解用户行为动态，还能生成动态推荐，提供动态适应的用户体验。



Temporal Neighbors Information Aggregation

在DynLLM框架中，我们利用时间图注意力网络（TGANs）来处理用户和物品的时序关系。具体来说，我们首先定义用户 u 在时间 t 的邻域 $\mathcal{N}_u(t)$ ，包括与之有交互的所有用户，包括自身以避免循环连接。然后，我们计算每个用户 u_i 的权重：

$\alpha_{i|u,t}^{(TGAN)} = \text{TGANs根据}(u, u_i, t) \text{计算的注意力系数，它体现了用户}$

u 对该邻居 u_i 在时间 t 的关注程度。接着，通过加权求和，我们得到用户 u 的时态邻域嵌入：

$$\mathbf{h}_u^{\mathcal{N}} = \sum_{u_i \in \mathcal{N}_u(t)} \alpha_{i|u,t}^{(TGAN)} \cdot \mathbf{h}_{u_i}$$

其中 \mathbf{h}_{u_i} 是用户 u_i 的原始嵌入，这个过程确保了用户行为的时空相关性得以整合。对于物品，计算方法与用户相同，只是用物品的嵌入替换用户嵌入。这样，这些邻域嵌入为用户提供个性化的动态推荐，既考虑了时间演变，又保持了对用户当前兴趣的洞察。

$$\mathcal{N}_u(t) = \{(i_1, t_1), (i_2, t_2), \dots, (i_n, t_n)\}.$$

在 DynLLM 模型中，为了将用户 u 在前一层次 $l-1$ 时间 t^- 的精确邻域内的物品嵌入

$$\{\mathbf{h}_{i_1}^{l-1}(t^-), \mathbf{h}_{i_2}^{l-1}(t^-), \dots, \mathbf{h}_{i_n}^{l-1}(t^-)\}$$

融合到当前时间 t 的邻域嵌入 $\mathbf{h}_u^{\mathcal{N}}(t)$

我们使用TGANs。TGANs 是一个通过参数 θ 学习的模型，它根据用户 u 的当前状态，以及其邻域中的物品嵌入，结合时间 t 的环境信息来动态计算注意力权重。具体操作如下：

$$\mathbf{h}_u^{\mathcal{N}}(t) = \text{TGAN}_{\theta}(\{\mathbf{h}_{i_1}^{l-1}(t^-), \mathbf{h}_{i_2}^{l-1}(t^-), \dots, \mathbf{h}_{i_n}^{l-1}(t^-), \mathbf{h}_u\}, t)$$

这里 TGAN_{θ} 对输入的用户和物品嵌入进行处理，通过学习的参数 θ 来计算适应性的权重，以平衡不同邻居对用户行为的影响。这样 $\mathbf{h}_u^{\mathcal{N}}(t)$ 就是用户 u 在时间 t 的动态邻域嵌入，为个性化推荐提供实时且全面的上下文信息。

$$\begin{aligned} \mathbf{e}_k^{(l-1)} &= \mathbf{h}_{i_k}^{(l-1)}(t^-) \parallel \phi(t - t_k), \\ \mathbf{K}^{(l)}(t) &= \mathbf{V}^{(l)}(t) = \mathbf{e}_1^{(l-1)} \parallel \mathbf{e}_2^{(l-1)} \parallel \dots \parallel \mathbf{e}_n^{(l-1)}, \\ \mathbf{q}^{(l)}(t) &= \mathbf{h}_u^{\mathcal{N},(l-1)}(t) \parallel \phi(0), \\ \mathbf{h}_u^{\mathcal{N},(l)}(t) &= \text{Attn}^{(l)}(\mathbf{q}^{(l)}(t), \mathbf{K}^{(l)}(t), \mathbf{V}^{(l)}(t)). \end{aligned}$$

在 DynLLM 模型中，初始的用户和物品邻域嵌入通常是在用户行为数据的基础上定义的。对于用户 u ，初始的邻域嵌入可能包含其在前一层次 $l-1$ 的所有历史交互中的物品嵌入，如

$$\mathbf{h}_{i_1}^{l-1}(t^-), \mathbf{h}_{i_2}^{l-1}(t^-), \dots, \mathbf{h}_{i_n}^{l-1}(t^-)$$

对于物品，它们的初始邻域嵌入可能是基于它们在当前时间 t 或与用户 u 相关的最近交互的嵌入。这些嵌入可能来自先前的推荐系统或者通过一些预训练方法得到。

在 TGANs 过程中，这些初始嵌入被用来初始化模型，然后通过学习的过程，TGANs 根据时间和用户行为更新这些嵌入，以生成动态的邻域嵌入 $\mathbf{h}_u^{\mathcal{N}}(t)$

以便为后续的个性化推荐服务。

$$\phi(t) = \sqrt{\frac{1}{d}} [\cos(\omega_1 t), \sin(\omega_1 t), \dots, \cos(\omega_d t), \sin(\omega_d t)],$$

在多层 TGANs（共 L 层）中，用户 u 的最终邻域嵌入为 $\mathbf{h}_u^{\mathcal{N}}(t) = \mathbf{h}_u^{\mathcal{N},L}(t)$

而物品 i 的则为 $\mathbf{h}_i^{\mathcal{N}}(t) = \mathbf{h}_i^{\mathcal{N},L}(t)$

接下来，我们引入一组可学习的参数 $\omega_1, \omega_2, \dots, \omega_d$ 和映射维数 d ，用于构建带有时间信息和社交网络结构的时空图嵌入。计算方法是：

$$\begin{aligned} \mathbf{h}_u^G(t) &= \text{Agg}_{\mathcal{N}_u(t)}(\{\mathbf{h}_u^{\mathcal{N},(l)}(t), l = 1 \dots L\}, \omega) \quad \text{and} \\ \mathbf{h}_i^G(t) &= \text{Agg}_{\mathcal{N}_i(t)}(\{\mathbf{h}_i^{\mathcal{N},(l)}(t), l = 1 \dots L\}, \omega) \end{aligned}$$

这里 **Agg** 代表一种将各个层次的邻域信息聚合的方法，它结合了时间和位置特征，通过 ω 融合，生成了全面的时空图嵌入，捕捉了用户和物品在时间上的行为动态以及他们在社交网络中的位置关系。

在这个场景中 $\mathbf{h}_u^p(t)$ 指的是一个用于处理交互历史不及时问题的投影嵌入，旨在减少即时交互事件对嵌入更新的影响。

$\mathbf{h}_i(t^-)$ 代表的是在时间 t^- 时对物品的动态嵌入，它记录了物品随时间的动态变化。

LLM-based Multi-faceted Information Augmentation

$$\text{crd}_u^{t-\Delta t, t}, \text{int}_u^{t-\Delta t, t}, \text{ctg}_u^{t-\Delta t, t}, \text{brd}_u^{t-\Delta t, t} = \text{LLM}_G(\mathcal{P}_u^{t-\Delta t, t}),$$

$$\begin{aligned} \mathbf{h}_u^{\text{crd}}(t) &= \text{MLP}(\text{LLM}_E(\text{crd}_u^{t-\Delta t, t})), \\ \mathbf{h}_u^{\text{int}}(t) &= \text{MLP}(\text{LLM}_E(\text{int}_u^{t-\Delta t, t})), \\ \mathbf{h}_u^{\text{ctg}}(t) &= \text{MLP}(\text{LLM}_E(\text{ctg}_u^{t-\Delta t, t})), \\ \mathbf{h}_u^{\text{brd}}(t) &= \text{MLP}(\text{LLM}_E(\text{brd}_u^{t-\Delta t, t})). \end{aligned}$$

Multi-faceted Distilled Attention Mechanism

在将多维转换后的用户画像嵌入整合到动态用户嵌入中时，我们面临的一个挑战是如何有效去除 LLMs 可能引入的噪声，同时增强嵌入的表达能力。为此，我们采用蒸馏技术。首先，我们针对用户画像嵌入 $\mathbf{h}_u^{\text{crd}}(t)$

定义一个参数向量 \mathbf{q} 来量化各维度的重要性。然后，我们选取排名前 r 的重要维度，对这些维度进行蒸馏，这里的 r 是蒸馏系数。蒸馏过程通过 **Distill** 函数实现，得到：

$$\hat{\mathbf{h}}_u^{\text{crd}}(t) = \text{Distill}(\mathbf{h}_u^{\text{crd}}(t), \mathbf{q}, r)$$

这样 $\hat{\mathbf{h}}_u^{\text{crd}}(t)$ 是经过处理后的、更纯净且信息丰富的动态用户嵌入，从而保证了推荐质量。

$$\begin{aligned} \mathbf{s} &= \mathbf{h}_u^{\text{crd}}(t) \mathbf{q} / \|\mathbf{q}\| \\ j &= \text{top-indices}(\mathbf{s}, r) \\ \mathbf{h}_u^{\text{crd}}(t) &= [\mathbf{h}_u^{\text{crd}}(t) \circ \tanh(\mathbf{s})]_j, \end{aligned}$$

接下来，我们将 LLM 生成的信息与基于图和时间的动态信息相结合。首先，我们将 LLM 增强的用户画像嵌入 $\mathbf{h}_u^A(t)$ 与用户的真实动态嵌入 $\mathbf{h}_u^G(t)$ 拼接，形成一个整体的融合嵌入。

$$\mathbf{h}_u^D(t) = \alpha \cdot \text{Attn}_{\text{multi-head}}(\mathbf{h}_u^A(t), \mathbf{h}_u^G(t)) + (1 - \alpha) \cdot \text{FFN}(\text{Drop}(\mathbf{h}_u^A(t)))$$

其中 α 是权重，通常设置在0到1之间，FFN用于深化特征处理。这样，动态用户嵌入能够综合LLM和图信息，以更好地反映用户行为的动态特性。

$$\begin{aligned} \mathbf{q}^{(l)}(t) &= \mathbf{h}_u^G(t), \\ \mathbf{h}_u^A(t) &= \mathbf{h}_u^{\text{crd}}(t) \parallel \mathbf{h}_u^{\text{int}}(t) \parallel \mathbf{h}_u^{\text{ctg}}(t) \parallel \mathbf{h}_u^{\text{brd}}(t), \\ \mathbf{K}^{(l)}(t) &= \mathbf{V}^{(l)}(t) = \mathbf{h}_u^A(t), \\ \mathbf{h}_u^{(l)}(t) &= \text{Attn}^{(l)}(\mathbf{q}^{(l)}(t), \mathbf{K}^{(l)}(t), \mathbf{V}^{(l)}(t)), \end{aligned}$$

在这个模型中，用户在时间 t 的动态嵌入是经过多层注意力计算后的最终输出，记作 $\mathbf{h}_u(t) = \mathbf{h}_u^{(L)}(t)$

它综合了图的结构信息、具体时间点的细节以及LLM增强的特征。而物品在时间 t 的嵌入则由两部分组成：一是对物品的平均表示 \mathbf{h}_i ，二是基于图信息的动态嵌入 $\mathbf{h}_i^G(t)$ 。这里的 $\mathbf{h}_i^G(t) = \mathbf{h}_i(t)$ 表示了物品在该时刻的实际动态状态。通过向量串联操作 \parallel ，我们保留了所有这些信息，形成物品的完整时间态表示 $\mathbf{h}_i(t)$ 。

Model Training and Update

在这个模型中，用户对物品的兴趣度计算公式为

$$\text{pref}(u, i, t) = \mathbf{h}_u(t) \cdot \mathbf{h}_i(t)$$

其中点积 \cdot 操作表示向量的相似度测量。这个相似度表示了用户和物品在特定时间 t 内表示的匹配程度，从而直接反映了用户对该物品的兴趣强度。

$$\hat{y}_{ui}^t = \mathbf{h}_u(t)^\top \mathbf{h}_i(t),$$

$$L(u, i, t) = \max(0, \text{pref}(u, i, t) - \text{pref}(u, j, t) + \Delta)$$

其中 Δ 是一个正偏差项，避免模型总是预测零评分。负样本 j 是与正样本 i 关联的一个随机选择。通过最小化这个损失，模型通过调整用户和物品的表示来优化预测，提升推荐的准确性和效果。

$$\ell = \sum_{(u, i^+, i^-, t) \in E} -\log \sigma(\hat{y}_{ui^+}^t - \hat{y}_{ui^-}^t) + \lambda \|\Theta\|^2.$$

这里， E 代表训练交互数据集， i^+ 和 i^- 分别是来自训练数据的正负样本。权重衰减正则化 $\|\Theta\|^2$ ，乘以系数 λ 用以防止过拟合，而 $\sigma(\cdot)$ 则是用于引入非线性的激活函数sigmoid⁺。为了进行下次推荐，我们使用GRU（门控循环单元）来整合当前时刻的交互信息，以更新用户和项目的嵌入表示，其计算方式为：

$$\begin{aligned} \mathbf{h}_u(t^+) &= \text{GRU}(\mathbf{h}_u(t), \mathbf{h}_i(t) \parallel \phi(t - t'_u)) \\ \mathbf{h}_i(t^+) &= \text{GRU}(\mathbf{h}_i^G(t), \mathbf{h}_u(t) \parallel \phi(t - t'_i)) \\ \mathbf{h}_i(t^+) &= \mathbf{h}_i \parallel \mathbf{h}_i(t^+). \end{aligned}$$

在模型中，我们用 t'_u 表示用户 u 最近互动的时间点， t'_i 表示物品 i 最后被用户互动的时间点。利用这些时间戳，我们更新用户和物品的嵌入向量。具体来说：

$$\mathbf{h}_u(t^+) = \text{updated}(\mathbf{h}_u, t'_u), \quad \mathbf{h}_i(t^+) = \text{updated}(\mathbf{h}_i, t'_i)$$

这里的'updated'函数是对原始嵌入向量进行相应时间信息的处理，以反映其时效性。这些更新后的向量将作为未来推荐的依据。

Experimental Settings

Data Description

Dataset	Tmall	Alibaba
Number of users	5651	16901
Number of items	10333	30080
Number of interactions	258390	722716
The average length of a title	35.51	34.22
The average interactions of a user	45.72	42.76
The average interaction of an item	25.51	24.03

实验设计如下：

1. 数据采集：我们从两个电子商务平台收集了两年的购买数据，时间区间为2021年11月至2023年11月，包含用户与商品的交互记录，包括每次交互的时间戳。
2. 数据预处理：我们随机抽取一部分用户和商品，满足购买多样性条件（至少20种商品的用户和5种商品的物品），并处理缺失值和异常数据。时间戳信息用于跟踪行为随时间的变化。
3. 特征工程：基于历史交互，我们构建用户和商品的动态特征，如最近购买行为、购买频率等。
4. 实验设置：我们构建了基线模型，如静态图推荐模型，同时对比 DynLLM 模型。所有模型在相同数据预处理和特征基础上进行训练和测试，确保公平性。
5. 评估指标：我们采用精确率、召回率、NDCG等推荐系统常用的评估指标，同时关注模型在处理大量实时数据时的效率，包括内存消耗和计算复杂度。

Baselines

在我们的研究中，我们对比了包括Matrix Factorization (MF)、Collaborative Filtering (CF)和Content-based filtering (CB)在内的静态图推荐方法，以及Dynamic Matrix Factorization (DMF)和Graph Convolutional Networks (GCN)、Graph Neural Networks (GNN)等动态图推荐模型。所有基线模型在相同的数据预处理和特征构建步骤下进行训练和测试，以保证公正性。我们通过精确率、召回率、NDCG等指标来衡量其在处理动态用户和商品信息时的性能。我们的目标是通过实证分析，证实DynLLM利用其特有的动态学习能力，能在实时数据流中显著超越这些竞争对手，提供更精准的推荐。

Evaluation Protocols

在Top-K推荐任务中，我们对DynLLM模型的效果进行了评估，采用了NDCG（折现累计增益）和召回率两个主流指标。在K=10、20、30的情况下，我们分别评估了 DynLLM 和基线的Recall@K和NDCG@K得分。为了保证全面性，我们按照时间顺序对数据集进行了70%训练、15%验证和15%测试的划分。在验证阶段，我们采用标准的早停策略，即如果验证集的Recall@10在连续10个epoch内无提升，就会停止训练。

Overall Performance (RQ 1)

DynLLM的创新性体现在其结合了动态学习（LLM）和图模型，通过这种动态图处理方式，它在捕捉用户和物品的动态交互模式上表现出优越性，特别是在处理非均匀数据和实时变化时。尽管DyRep和DynShare在某些方面可能有局限，但DynLLM的这种综合优势使其在Top-K推荐任务中具有显著的竞争力。

Dataset	Tmall						Alibaba					
Model	R@10	R@20	R@30	N@10	N@20	N@30	R@10	R@20	R@30	N@10	N@20	N@30
GCN	0.0287	0.0381	0.0610	0.0218	0.0241	0.0290	0.0159	0.0322	0.0540	0.0087	0.0127	0.0174
GAT	0.0283	0.0489	0.0636	0.0189	0.0240	0.0271	0.0233	0.0351	0.0463	0.0141	0.0170	0.0194
NGCF	0.0594	0.0999	0.1267	0.0364	0.0465	0.0522	0.0510	0.0732	0.0927	0.0321	0.0377	0.0418
LightGCN	0.0703	0.1105	0.1442	0.0405	0.0506	0.0578	0.0458	0.0697	0.0864	0.0296	0.0355	0.0391
Jodie	0.0270	0.0516	0.0732	0.0118	0.0179	0.0225	0.0253	0.0528	0.0783	0.0107	0.0176	0.0230
DyRep	0.0450	0.0826	0.1131	0.0213	0.0307	0.0372	0.0455	0.0899	0.1242	0.0199	0.0310	0.0383
TGN	0.0726	0.1276	0.1766	0.0338	0.0476	0.0580	0.0516	0.0923	0.1263	0.0248	0.0350	0.0422
DynShare	0.0750	0.1279	0.1718	0.0358	0.0491	0.0584	0.0563	0.1039	0.1440	0.0252	0.0371	0.0457
DynLLM	0.1052	0.1765	0.2321	0.0501	0.0680	0.0799	0.0685	0.1120	0.1459	0.0340	0.0449	0.0521

2. LLM通过生成多维度用户描述，它通过深度学习丰富了用户信息，同时蒸馏注意力机制整合了LLM增强信息与时间图信息，确保了信息的有效整合。

3.LLM生成的多维描述在蒸馏过程中保留了关键信息，减少了噪声，增强了DynLLM对复杂动态交互模式的理解和处理能力，提高了框架的鲁棒性。

动态图推荐在预测用户喜好方面通常显著优于静态方法，强调了时间动态的重要性。然而，Jodie在某些场景下不敌NGCF和LightGCN，尽管它们考虑了时间因素，但未能充分捕捉高阶图结构。相比之下，TGN和DynShare通过整合时序与高阶信息超越了仅依赖静态图的模型。尽管DyRep因为缺乏记忆模块处理历史信息稍显不足，总体上它们在理解和利用动态关系方面表现出优势。

在静态图推荐中，NGCF和LightGCN通过引入高阶连通性的概念，如GCN和GAT等方法，显著提升了推荐效果，超越了基础的图模型。这表明在推荐系统中，精确地捕捉用户-项目间的复杂交互关系对于信号建模至关重要，高阶连接性是提升预测准确性的重要因素。

DynShare凭借其独特的内存管理和非线性投影器，展现出在预测用户嵌入和防止过时方面的优势，超越了其他动态图模型。至于我们的DynLLM，它在众多动态方法中脱颖而出，证实了LLMs在揭示用户隐含属性的强大能力，显著提升了动态用户嵌入的表达质量和时效性。

原文《DynLLM: When Large Language Models Meet Dynamic Graph Recommendation》

发布于 2024-06-05 10:44 · IP 属地北京

阿里巴巴工作 LLM 推荐系统



理性发言，友善互动



还没有评论，发表第一个评论吧

推荐阅读

互联网裁员！大厂优秀者来量化吧！高薪招聘岗位！

高薪聘请2021/2022届本/硕/博 数学、物理、统计、计算机、软件等专业 通过这里找到我 SLam_Pan 1、量化软件开发工程师 (本科211以上，硕士学位)base北上杭深 关键词:c++, python, java软件... 量化猎头SLam

国内首位！Node.js社区将阿里云工程师张秋怡吸纳为CTC成员

继去年MariaDB基金会将 阿里云数据库高级专家彭立勋列为个人成员后，今年又有一位阿里云工程师在国际技术社区重要工作中有所担当，她就是张秋怡。张秋怡，花名洗影，阿里云云应用服务团队工... 阿里云云栖... 发表于程序员进修...



InfoQ 大咖说 | 对话阿里巴巴 贾扬清：如何成为一名优秀的...

InfoQ

阿里、腾讯、百度、快手 然语言处理实习生面经

两年制硕士，今年春季开始去实习，只有一个项目，一个本科时大厂数据分析的经历，无论文，直到5月24才等来的意向书，按照时间顺序，下具体面试内容，我就是人.. Behind