

赞同 39



分享

## 华为2024：ClickPrompt——巧借CTR模型之力，赋能LLM精准预测点击率



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

39 人赞同了该文章

### Introduction

**点击率<sup>+</sup> (CTR)** 预测是互联网领域中的关键任务，它旨在通过分析用户行为数据，预测用户对某个特定内容或广告的点击概率。

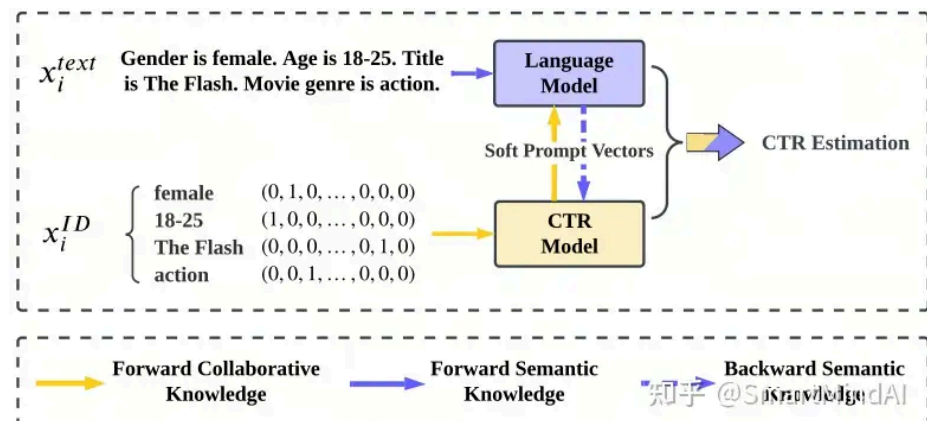
十年来，CTR模型不断进化以捕获交互特征，但面临主要问题-----'语义信息丢失'。由于文本数据经独热编码转化为ID，忽略了原始数据中的关联信息。例如，电影《复仇者联盟4》不仅与前作关联，还通过潜在语义与其他超级英雄电影相关。然而，这种转换导致这些语义信息在冷启动、低频特征及点击信号弱时预测表现下降。为解决此问题，近来研究开始利用预训练语言模型

(PLMs)。通过将文本数据转化为文本特征，而非ID，模型能更好地保留并利用这些语义信息。对于每个输入样例 $x_i$ ，创建了相应的文本特征表示。

点击率预测中，尽管预训练语言模型 (PLMs) 在理解语义和学习知识上具备优势，却面临着两个主要难题：预测精确度不高和计算效率低下。首先，纯ID特征缺乏语义，如ID 'AX529' 分解成无效的词汇序列，PLMs无法充分利用其内在含义。其次，线性组合ID特征时，模型难以捕捉深层交互，丢失了至关重要的特征级交互信息。

为了提高效率，研究尝试通过添加嵌入表、优化适配器或改进索引策略来优化，但对如何利用ID中的协作知识仍处于探索初级阶段。然而，这种方法往往伴随着高昂的计算成本和内存需求，尤其是当使用大模型时，这限制了其在对实时性要求高的在线服务中的应用，因为响应速度通常需要几十毫秒甚至更短。

为克服这个问题，一些研究转向预先缓存PLM的输出以减少计算，但这可能导致推荐系统的实时性受损，影响预测效果。ClickPrompt框架旨在优化点击率预测，同时解决预训练语言模型带来的推理效率问题。它将CTR模型与预训练语言模型结合，形成一个软提示生成器。



PLM，同时保持文本的语义信息回流至CTR模型。ClickPrompt采用预训练-微调策略，通过两种微调策略对CTR预测进行优化，以实现高效预测的同时减小计算成本。

- **微调与PLM整合**：我们的策略是直接将CTR模型与预训练语言模型作为一个联合系统进行微调。通过在二者之间添加一个提示生成层，这个层负责生成针对ID特征的协作软提示，同时接收PLM的语义信息。这种互动使得两种知识能够互补，协同提升点击率预测的精准度。
- 另一种解决方案是不依赖PLM微调。通过利用PA-MLM这样的预训练模型已有的语义理解能力，我们直接对CTR模型进行微调。这样既能保留模型结构，又能通过预训练的语义感知初始化提高性能，而且无需额外增加计算负担，保证了实时性。
- 我们构建了ClickPrompt框架，将CTR模型作为生成软提示的核心，转变为PLMs的辅助工具。通过PA-MLM预训练，我们不直接改变PLM，而是让它生成针对ID特征的提示，这样既保留了语义信息，又减少了微调过程中的计算负担。ClickPrompt设计灵活，适用于各种CTR模型，经实证验证，它在提升预测准确性和优化推理效率方面表现出优越性。
- ClickPrompt设计巧妙，它将CTR模型转化为生成软提示的源，与PLMs协同工作。通过预训练的PA-MLM任务，模型能有效整合ID特征的协作知识与文本的语义信息，而无需微调PLMs。这种方法保证了预测精度的同时，减少了计算成本，体现了模型的通用性和实用性。无论是何种CTR模型或PLMs，ClickPrompt都能无缝对接，实现高效且准确的点击率预测。
- 我们运用ClickPrompt在四个实际世界中的公开数据集上进行了全面的实验，结果显示，相较于现有基准模型，我们的方法显著提升了点击率预测的准确性。这进一步验证了ClickPrompt的有效性和优越性。

## Preliminaries

### Traditional CTR Prediction

点击率预测可以理解对包含F个特征的多分类二元分类问题，每个特征对应一个唯一类别。具体到模型，每个样本 $(\mathbf{x}_i, y_i)$ 表示，其中 $\mathbf{x}_i$ 通过独热编码转化为ID特征向量 $\mathbf{x}_i^{ID}$ 。目标是预测每个样本 $P(y_i = 1|\mathbf{x}_i)$ ，即点击概率。传统的神经CTR模型，如大多数，由三个主要层次构成。

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^N [y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)]$$

在这个上下文里， $N$ 代表训练样本集的大小，用于表示用于模型训练的数据点数量。

### PLM-based CTR Prediction

在利用PLMs处理点击率估计问题时，我们借鉴了模型的语义关联性。不同于传统的CTR预测，我们不是简单地输入 $\mathbf{x}_i$ ，而是将其转化为文本格式 $\mathbf{x}_i^{text}$ ，如图中所示。根据任务需求和标签定义，PLM的CTR预测通常有两种主要方式：1. 二元文本分类：这类方法将预测目标设为二元输出（ $y_i \in \{0, 1\}$ ），保持与传统设置相同。首先，通过PLMs获取文本特征的密集表示 $\mathbf{q}_i^{text}$ ，然后用二元交叉熵<sup>+</sup>（BCE loss）作为损失函数进行优化。这种方法强调了PLM在理解和解析输入文本信息上的能力，以提高点击率预测的准确性。

$$\hat{y}_i = \sigma(\text{MLP}(\text{PLM}(\mathbf{x}_i^{text}))) \in (0, 1)$$

## Methodology

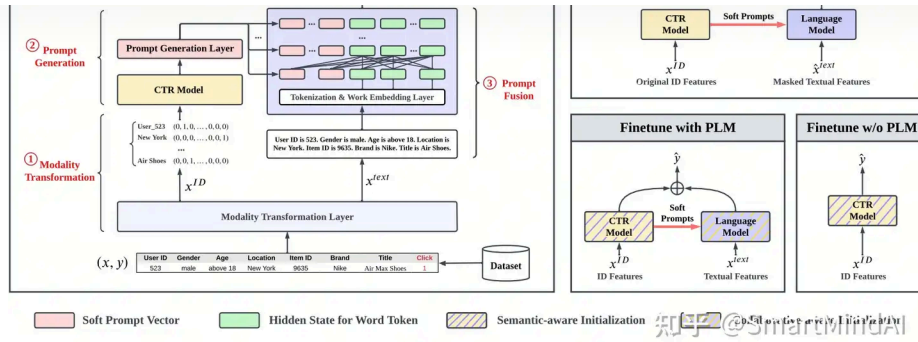


Figure 2: The illustration of the model architecture and learning strategy for our proposed ClickPrompt framework.

我们采用点击率预测作为核心任务，将CTR模型转化为生成软提示的PLM模型。我们通过PA-MLM进行预训练，利用ID特征生成的软提示与文本特征交互，让PLM理解和处理协同知识。在模型架构上，我们附加一个多层感知机<sup>+</sup>（MLP）到PLM生成的文本表示 $q_i^{text}$ ，以利用其丰富的语义信息。学习策略上，我们采用二元交叉熵损失（BCE loss）进行微调，确保预测结果的精确性，同时保持了与传统方法的兼容性，通过这种方式，我们实现了对预测效率和准确性的提升。

## Overview of ClickPrompt

1. 数据预处理：首先，将原始的点击率样本 $(x_i, y_i)$ 转换为ID特征向量 $x_i^{ID}$ ，这是通过独热编码完成的。
2. 特征融合<sup>+</sup>：接着，利用预训练语言模型（PLM）处理ID特征，生成与文本序列相关的软提示 $x_i^{text}$ 。这一步是模型的核心，通过PLM的语义理解能力，捕捉输入的深层含义。
3. 模型集成：生成的文本表示 $q_i^{text}$ 与ID特征相结合，通过一个多层感知机（MLP）进行融合，以利用PLM的丰富语义信息和ID特征的细节。

在学习过程中，我们使用二元交叉熵损失（BCE loss）作为目标函数，对整个模型进行微调，确保点击率预测的精度，同时保持与传统CTR模型的兼容性，优化了预测效率。

## Modality Transformation

$x_i^{text}$ 是在 ClickPrompt 模型中，对于每个训练样本 $i$ ，PLM生成的文本表示。它是通过对ID特征进行处理（如独热编码）后的结果，用于与ID特征相结合，通过MLP来融合和提升点击率预测的准确性。

$$x_{i,j}^{text} = [f_j^{name}, < code > is ", f_{i,j}, < /code >."], j = 1, \dots, F,$$

$$x_i^{text} = [x_{i,1}^{text}, x_{i,2}^{text}, \dots, x_{i,F}^{text}],$$

在这个上下文中 $f_j^{name}$ 是第 $j$ 个特征的实际名称 $f_{i,j}$ 是第 $i$ 个数据样本的第 $j$ 个特征的具体数值。 $[<code>]$ 是用来连接列表中元素的符号，表示将所有特征名与对应的特征值用空格串联起来，形成一个描述样本特征的字符串，如

$$(f_1^{name}, f_2^{name}, \dots, f_F^{name}) \text{ 和 } (f_{1,1}, f_{1,2}, \dots, f_{1,F})$$

这样的表示方式常用于输入到模型中，以便于模型理解和处理。

## Prompt Generation

在提示生成阶段，我们的目标是将ID特征 $x_i^{ID}$ 转换为富含协作知识的非明确性向量。为了实现这一目标，我们利用CTR模型的嵌入（通过 $f_{CTR}$ ）和特征交互（FI）层，这个过程可以表示为：

$$q_i = f_{CTR}(x_i^{ID}, FI)$$

这里 $f_{CTR}$ 描述了CTR模型如何结合ID信息和内部学习，生成一种融合了语义信息和协作知识的紧凑表示，为后续融合做准备。这种向量包含了丰富的知识，能够支持更准确的点击率预测。

$$q_i = FI\_Layer(Embed\_Layer(x_i^{ID}))$$

做的目的是个性化地提取每个ID特征的语义信息，使其与上下文紧密相关，同时保持与原始ID特征的联系。

通过这些网络，模型能够从ID特征中提取深层次的语义信息，然后转化成适合与文本特征（ $q_i^{text}$ ）融合的形式。这种策略旨在增强预测的精确性，因为不同的特征可以通过互补的方式共同提升点击率预测的整体性能。

$$p_{i,l,k} = g_{l,k}(q_i), 1 \leq l \leq L, 1 \leq k \leq K$$

在这个表述中， $p_{i,l,k}$ 代表在PLM第 $l$ 层的第 $k$ 个软提示向量，PLM包含 $L$ 层，每层有 $K$ 个这样的向量。每个独立的投影网络 $g_{l,k}(\cdot)$ 是一个多层感知器<sup>+</sup>（MLP），用于对ID特征的相应软提示进行逐层处理。设计上的目标是保持维度一致性和空间变换的灵活性，这样网络可以深入挖掘ID特征的语义内容。

通过这种方式 $g_{l,k}$ 不仅能够提取ID特征的深层信息，还能将这些信息转化为能与文本特征（如 $q_i^{text}$ ）相融合的形式。ClickPrompt正是通过这种策略，巧妙地利用了ID特征和文本特征的互补性，从而显著提高了点击率预测的准确性和效率。

### Prompt Fusion

$$h_i^l = \text{Transformer}_l(x_i^{text}, p_{i,1:k_l})$$

其中 $h_i^l$ 是该层的输出 $k_l$ 表示选取的前 $k$ 个软提示。 $\text{Transformer}_l$ 是Transformer层的计算模块，它结合了词令牌和软提示，形成带有上下文语义的表示。ClickPrompt通过这种方式整合了PLM的上下文理解和文本特征的语义，以此提升点击率预测的精准度。

$$z_{z=1}^L = \text{Transformer}_L([p_{i,l,k}]_{k=1}^K \oplus [h_{i,l,z}]_{z=1}^Z)$$

在这个模型流程中，对于每个词令牌， $[h_{i,l,z}]_{z=1}^Z$ 表示在第 $l$ 层的隐藏状态。通过Transformer层的自注意力机制，CTR模型信号能够与文本特征的语义信息在软提示层面精确对齐和融合。经过 $L$ 层的迭代，PLM的最终输出包含了所有层级的信息。接下来，对所有隐藏状态进行池化<sup>+</sup>，以提取核心信息，然后通过预测层进行处理。这一阶段的目标是综合ID特征和文本特征的优势，从而产生更精确的点击率预测。ClickPrompt通过这种方式实现了两种模态的协同优化，以提升预测的准确性。

$$\text{MLP}(\text{Pooling}([h_{i,L+1,z}]_{z=1}^Z))$$

输出维度的选择、后续激活函数<sup>+</sup>和损失函数的具体选择会在下一节中根据实际任务需求进行详细讨论。这通常会依据模型性能、训练效果以及任务的复杂性来决定，旨在最大化模型在点击率预测上的表现。

### Learning Strategy

如图所示，ClickPrompt采用了经典的预训练-微调方法。首先，它引入了prompt-augmented masked language modeling(PA-MLM)作为一种预训练任务。这个任务通过巧妙地结合软提示与合作和语义信息，通过引导软提示与正文之间的互动来增强模型对知识的理解和联系，以此对模型进行初始化，优化参数。

接下来，ClickPrompt采用两种微调策略：

1. 联合微调：在预训练的基础上，直接将模型应用于点击率预测任务，让模型同时学习如何处理ID特征和文本特征，以获得更全面的上下文理解。
2. 交替微调：可能是在预训练阶段和微调阶段之间交替进行，先让模型专注于预训练任务，然后逐渐过渡到专门针对点击率的微调，以避免过早适应特定任务导致的泛化问题。

这两种微调策略都是为了提高模型在实际点击率预测任务中的性能，同时保持与基础预训练的兼容性和灵活性。

### Prompt-augmented Masked Language Modeling

解码器结构，后跟softmax函数和交叉熵损失<sup>+</sup>。

通过设置15%的随机概率，分别对每个输入执行(1) 替换为`< code > MASK < /code >`，(2) 随机单词替换，和(3) 保持不变的操作，PLM需从软提示中提取关键信息来填充遮盖位置。这种策略旨在促使PLM与CTR模型在细节级别上精确对齐，协同完成点击率预测任务。

Finetuning with PLM

当然，模型结构可保持不变，我们仍可在下文中对整个模型进行监督微调，以适应CTR预测任务。如图所示，模型架构中，CTR模型和PLM之间通过软提示向量实现了深度集成。它们之间进行了直接的交互，利用了PLM对ID特征的语义理解，同时利用软提示来补充和增强CTR模型对文本特征的处理。这种结合使得模型能更好地理解 and 预测点击行为，从而提升预测精度。

$$\begin{aligned} \hat{y}_i^{CTR} &= \text{MLP}(q_i) \in \mathbb{R}, \\ \hat{y}_i^{PLM} &= \text{MLP}\left(\text{Pooling}\left([h_{i,L+1,z}]_{z=1}^Z\right)\right) \in \mathbb{R}, \\ \hat{y}_i &= \sigma\left(\alpha \times \hat{y}_i^{CTR} + (1 - \alpha) \times \hat{y}_i^{PLM}\right), \end{aligned}$$

在这个模型设置中，参数 $\alpha$ 是个可调节的权重，它决定了ID特征和文本特征在预测中的相对重要性。 $\text{sigmoid函数}^+\sigma(\cdot)$ 确保了输出在0到1之间，便于量化和组合。在微调阶段，这个过程使得ID特征和文本特征的交互更加紧密，它们的语义知识通过软提示相互融合，形成一个有机的整体。通过这种动态权重和功能的结合，ClickPrompt能够在预测过程中实现协同优化，显著提升CTR（点击率）的预测准确性。

Finetuning without PLM

为了解决效率问题，我们提出仅对专门的CTR模型进行微调，而非同时优化PLM。在PA-MLM预训练阶段，我们通过反向传播途径，巧妙地将PLM的语义知识注入到CTR模型中，这是在保持其结构不变且不增加额外计算负担的前提下，实现两者知识融合的关键。这种方法利用了语义意识的参数初始化，使得模型能够在不额外训练的情况下，通过隐性交互提升CTR预测的准确性和效率。

$$\hat{y}_i = \sigma\left(\text{MLP}(q_i)\right) \in \mathbb{R}$$

对于这两种微调策略，我们均使用二元交叉熵损失（如等式所示）来量化点击概率的估计。这种损失函数在计算上关注于模型对真实点击事件的正确分类，通过优化这个指标，确保模型能够准确预测点击行为。

Experiment Setups

Datasets

我们选择了四个代表性推荐场景的公开数据集，包括MovieLens-1M、BookCrossing、Amazon-Toys和GoodReads，它们分别来源于不同来源（见注释）。这些数据集以8:1:1的比例被划分为训练集、验证集和测试集<sup>+</sup>。

Table 1: The dataset statistics

Dataset	#Training	#Validation	#Test	#Fields	#Features
Movielens-1M	591,208	73,902	73,902	8	17,251
BookCrossing	824,936	103,117	103,118	8	722,234
Amazon-Toys	1,489,782	186,223	186,223	5	371,813
GoodReads	16,097,632	2,012,204	2,012,204	15	4,565,450

Evaluation Metrics

为了衡量ClickPrompt的性能，我们使用AUC（Receiver Operating Characteristic曲线下面积）和Log Loss（二元交叉熵损失）作为评估标准。AUC代表了模型在区分真实点击与非点击上的能



Baselines

对于传统的CTR模型，我们构建了多样的基线，包括FM、DNN、DeepFM、xDeepFM、PNN、DCN、AutoInt、FiGNN、FGCNN和DCNv2，这些模型通过不同的特征交互策略来处理输入数据。

而对于基于预训练语言模型的CTR，我们选取了具有代表性的基线，如CTR-BERT、P5、PTab和CTRL，以评估其在CTR预测任务中的性能。这些模型利用了PLM的潜在语义信息，与ClickPrompt进行对比，旨在考察其在这一领域的能力和优势。

Implementation Details

在优化过程中，我们选用AdamW作为优化器，对基于PLM的预训练模型设置批量大小1024，学习率为5e-5。学习率的warm-up比例从0%、5%、10%三个选项中选择。预训20轮。针对各数据集，微调时批大小分别为256（MovieLens-1M）、256（BookCrossing）、1024（Amazon-Toys）和4096（GoodReads）。对于CTR模型部分，学习率固定为1e-3，而PLM部分的学习率从0（冻结）、3e-5和5e-5三个选项中挑选。提示生成的投影网络由一个双层tanh激活的多层感知机构成，其隐藏层大小等于PLM嵌入大小。提示的数量K从1、3、5、7四个选项中选择。由于ClickPrompt是模型无关的，我们默认用DCNv2作为CTR模型，RoBERTa-base作为PLM，除非有特别说明。在测试阶段，我们依据验证集上的AUC最优点来评估模型性能。

Overall Performance (RQ1)

Table 2: The overall performance comparison of different models. The best result is given in bold, while the second-best value is underlined. We also use wavy underline to denote the best baseline performance. Rel.Impr denotes the relative AUC improvement rate of our proposed ClickPrompt<sub>with PLM</sub> against each baseline model. The symbol \* indicates statistically significant improvement of ClickPrompt over the best baseline model with  $p < 0.001$ .

Model	MovieLens-1M			BookCrossing			Amazon-Toys			GoodReads		
	AUC	Log Loss	Rel.Impr	AUC	Log Loss	Rel.Impr	AUC	Log Loss	Rel.Impr	AUC	Log Loss	Rel.Impr
FM	0.8371	0.4090	1.53%	0.7871	0.5202	2.02%	0.6668	0.4059	1.30%	0.7614	0.5190	1.85%
DNN	0.8413	0.3944	1.02%	0.7940	0.5124	1.13%	0.6686	0.3982	1.03%	0.7685	0.5082	0.91%
DeepFM	0.8443	<u>0.3915</u>	0.66%	0.7959	0.5106	0.89%	0.6692	0.3978	0.94%	0.7690	0.5136	0.85%
xDeepFM	0.8435	0.3950	0.76%	0.7943	0.5122	1.10%	0.6681	0.3967	1.11%	0.7697	0.5072	0.75%
IPNN	0.8437	0.3926	0.73%	0.7953	0.5111	0.97%	0.6687	0.3980	1.02%	0.7722	0.5148	0.43%
DCN	0.8423	0.3964	0.90%	0.7952	0.5116	0.98%	0.6688	0.3964	1.00%	0.7693	0.5074	0.81%
AutoInt	0.8399	0.4004	1.19%	0.7954	0.5113	0.96%	0.6678	0.3977	1.15%	0.7682	0.5084	0.95%
FiGNN	0.8399	0.3991	1.19%	0.7970	0.5105	0.75%	0.6700	<u>0.3947</u>	0.82%	0.7667	0.5094	1.15%
FGCNN	0.8416	0.3957	0.99%	0.7985	<u>0.5082</u>	0.56%	0.6675	0.3978	1.20%	0.7705	0.5064	0.65%
DCNv2	0.8439	0.3954	0.71%	0.7970	0.5096	0.75%	0.6701	0.3961	0.81%	0.7711	0.5059	0.57%
CTR-BERT	0.8296	0.4208	2.45%	0.7848	0.5268	2.32%	0.6649	0.3988	1.59%	0.7457	0.5292	4.00%
P5	0.8173	0.4171	3.99%	0.7695	0.5360	4.35%	0.6470	0.4018	4.40%	0.7367	0.5531	5.27%
PTab	0.8353	0.4081	1.75%	0.7979	0.5208	0.64%	0.6685	0.3995	1.05%	0.7566	0.5203	2.50%
CTRL	<u>0.8453</u>	0.3932	0.54%	<u>0.7992</u>	0.5092	0.48%	<u>0.6704</u>	0.3960	0.76%	<u>0.7735</u>	<u>0.5038</u>	0.26%
ClickPrompt <sub>w/o PLM</sub>	<u>0.8467</u> *	0.3939	-	<u>0.8013</u> *	<u>0.5051</u> *	-	<u>0.6719</u> *	<u>0.3933</u> *	-	<u>0.7744</u> *	<u>0.5034</u> *	-
ClickPrompt <sub>with PLM</sub>	<b>0.8499</b> *	<b>0.3905</b> *	-	<b>0.8030</b> *	<b>0.5037</b> *	-	<b>0.6755</b> *	<b>0.3890</b> *	-	<b>0.7755</b> *	<b>0.5022</b> *	-

- ClickPrompt以其在AUC和LogLoss上的优异表现。
- ClickPrompt具有良好的兼容性，能与多种CTR模型和预训练语言模型配合，展现其广泛应用的可能性。
- 配置选择对ClickPrompt性能有显著影响，比如不同的学习率和提示数量K，优化时需找到最佳组合。
- 虽未直接针对长尾低频用户，但ClickPrompt的泛化能力可能强，能适应多样化输入，包括数据稀疏和多样性的场景，但具体表现需通过实验数据证实。

Model Compatibility (RQ2)

The improvements are statistically significant with  $p < 0.001$  against the corresponding raw CTR models (i.e., w/o).

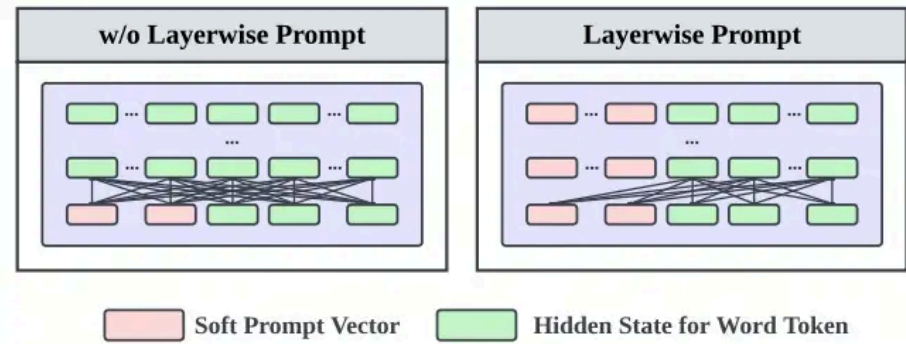
CTR Model	Finetuning	Language Model	Movielens-1M			BookCrossing			Amazon-Toys		
			AUC	Log Loss	RelImpr	AUC	Log Loss	RelImpr	AUC	Log Loss	RelImpr
DCNv2	w/o PLM	N/A	0.8439	0.3954	-	0.7970	0.5096	-	0.6701	0.3961	-
		TinyBERT	0.8464	0.3943	0.30%	0.7997	0.5070	0.34%	0.6705	0.3956	0.06%
		RoBERTa-base	0.8467	0.3939	0.33%	0.8013	0.5051	0.54%	0.6719	0.3933	0.27%
	with PLM	RoBERTa-large	<b>0.8476</b>	<b>0.3920</b>	0.44%	<b>0.8017</b>	<b>0.5047</b>	0.59%	<b>0.6723</b>	<b>0.3939</b>	0.33%
		TinyBERT	0.8470	0.3933	0.37%	0.8003	0.5063	0.41%	0.6732	0.3943	0.46%
		RoBERTa-base	<b>0.8499</b>	<b>0.3905</b>	0.71%	<b>0.8030</b>	<b>0.5037</b>	0.75%	<b>0.6755</b>	<b>0.3890</b>	0.81%
AutoInt	w/o PLM	RoBERTa-large	0.8498	0.3918	0.70%	<b>0.8032</b>	<b>0.5034</b>	0.78%	<b>0.6759</b>	<b>0.3893</b>	0.87%
		N/A	0.8399	0.4004	-	0.7954	0.5113	-	0.6678	0.3977	-
		TinyBERT	0.8422	0.3995	0.27%	0.7967	0.5098	0.16%	0.6714	0.3948	0.54%
	with PLM	RoBERTa-base	0.8439	0.3967	0.48%	0.7981	0.5091	0.34%	0.6724	0.3944	0.69%
		RoBERTa-large	<b>0.8454</b>	<b>0.3965</b>	0.65%	<b>0.7989</b>	<b>0.5084</b>	0.44%	<b>0.6732</b>	<b>0.3918</b>	0.81%
		TinyBERT	0.8458	0.3915	0.70%	0.7981	0.5081	0.34%	0.6728	0.3943	0.75%
DNN	w/o PLM	RoBERTa-base	0.8465	0.3912	0.79%	0.8004	0.5076	0.63%	0.6760	0.3924	1.23%
		RoBERTa-large	<b>0.8481</b>	<b>0.3893</b>	0.98%	<b>0.8009</b>	<b>0.5070</b>	0.69%	<b>0.6767</b>	<b>0.3893</b>	1.33%
		N/A	0.8413	0.3944	-	0.7940	0.5124	-	0.6686	0.3982	-
	with PLM	TinyBERT	0.8435	0.3944	0.26%	0.7960	0.5114	0.25%	0.6700	0.3956	0.21%
		RoBERTa-base	0.8448	0.3929	0.42%	0.7972	0.5097	0.40%	0.6704	0.3943	0.27%
		RoBERTa-large	<b>0.8455</b>	<b>0.3927</b>	0.50%	<b>0.7985</b>	<b>0.5081</b>	0.57%	<b>0.6710</b>	<b>0.3942</b>	0.36%
DNN	with PLM	TinyBERT	0.8446	0.3925	0.39%	0.7971	0.5093	0.39%	0.6732	0.3946	0.69%
		RoBERTa-base	0.8455	<b>0.3909</b>	0.50%	0.7994	0.5080	0.67%	0.6737	0.3947	0.67%
		RoBERTa-large	<b>0.8462</b>	0.3914	0.58%	<b>0.7999</b>	<b>0.5070</b>	0.74%	<b>0.6745</b>	<b>0.3930</b>	0.88%

1. 广泛兼容：ClickPrompt具备出色的兼容性，能无缝集成到各种基础的CTR模型（如DCNv2、AutoInt和DNN）以及不同层次的预训练语言模型（如TinyBERT、RoBERTa-base和RoBERTa-large），证明其适应性强。
2. 效果稳定：无论模型大小，ClickPrompt都能稳定提升性能，这说明其对模型架构的调整对其效果影响较小，体现了稳健性。
3. 规模效应：随着预训练模型规模的增加（从TinyBERT到RoBERTa-large），ClickPrompt的性能提升趋势保持一致，显示出对大模型资源的有效利用。
4. 强化语义理解：虽然没有直接提及具体通过什么方式，但ClickPrompt能够利用预训练的语义理解，即使不直接使用PLM也能显著提升预测准确性，暗示其在理解上下文信息方面有独特优势。

Ablation Study (RQ3)

我们研究了ClickPrompt的关键参数和配置如何影响其表现，重点关注提示设计和知识融合策略。实验中，我们以DCNv2、AutoInt和DNN这两类基础CTR模型，以及RoBERTa-base作为PLM基础，应用于MovieLens-1M、BookCrossing和Amazon-Toys数据集，通过finetuning-with-PLM策略来检验其有效性。

Prompt Strategy



**Figure 3: The illustration of prompt strategy variants. We can either only insert the prompts at the first input layer (left), or perform layerwise soft prompts for PLM (right). Note that ClickPrompt adopts the latter prompt strategy.**

我们对比了图中的两种不同prompt策略，并在表格中给出了相应的结果。结果显示，层级提示策略明显优于无层级提示策略。当仅在浅层输入层加入提示向量时，尽管可能利于增强协同知识，但过早的互动可能导致PLM的语义知识处理受损，从而影响两者间平衡，进而降低CTR模型的表现。

Dataset	CTR Model	w/o Layer-wise		Layer-wise		Relatp
		AUC	Log Loss	AUC	Log Loss	
Movielens-1M	DCNv2	0.8468	0.3948	<b>0.8499</b>	<b>0.3905</b>	0.37%
	AutoInt	0.8445	0.3946	<b>0.8465</b>	<b>0.3912</b>	0.24%
	DNN	0.8433	0.3959	<b>0.8455</b>	<b>0.3909</b>	0.26%
BookCrossing	DCNv2	0.7993	0.5075	<b>0.8030</b>	<b>0.5037</b>	0.46%
	AutoInt	0.7982	0.5091	<b>0.8004</b>	<b>0.5076</b>	0.28%
	DNN	0.7981	0.5109	<b>0.7994</b>	<b>0.5080</b>	0.16%
Amazon-Toys	DCNv2	0.6712	0.3945	<b>0.6755</b>	<b>0.3890</b>	0.64%
	AutoInt	0.6702	0.4006	<b>0.6760</b>	<b>0.3924</b>	0.87%
	DNN	0.6695	0.3962	<b>0.6742</b>	<b>0.3935</b>	0.76%

Collaborative & Semantic Knowledge Fusion Strategy

在ClickPrompt中，关键的技术涉及到两个关键点，分别是协同知识的整合和语义知识的对齐。这两点是确保模型有效利用PLM语义信息并优化CTR预测的关键步骤。

- 1. 从架构视角，ClickPrompt通过逐层的软提示设计建立了CTR模型与PLMs间的直接沟通桥梁，促进了协同知识和语义信息的有效传递。
- 2. 从学习策略上看，PA-MLM预训练促使PLM学会从提示向量中提取和整合潜在协作信息，实现了对语义对齐的精细处理，这是通过细粒度的互动过程实现的。
- 3. 无提示基础 (Base): 这个版本不使用任何特殊提示，仅直接对PLM进行微调，作为基准，以衡量ClickPrompt改进的效益，它不包含ClickPrompt的特性。
- 4. 硬对齐 (HA): 我们设计了一个变体，强制性地协同信息内嵌到模型中，通过在输入中直接操作相关特征，以此对比ClickPrompt，看看能否达到相似效果。
- 5. 软对齐 (SA): 类似ClickPrompt，但不依赖PA-MLM预训练，而是通过其他方法（如注意力机制）在训练中间接引导语义和协同信息的融合，以此探究其在融合方面的差异。

Table 5: The ablation study on the fusion strategy for collaborative and semantic knowledge. The best values are given in bold, while the second best values are underlined.

CTR Model	Variant	Movielens-1M		BookCrossing		Amazon-Toys	
		AUC	Log Loss	AUC	Log Loss	AUC	Log Loss
DCNv2	ClickPrompt	<b>0.8499</b>	<b>0.3905</b>	<b>0.8030</b>	<b>0.5037</b>	<b>0.6755</b>	<b>0.3890</b>
	w/o Prompt	<u>0.8470</u>	<u>0.3939</u>	<u>0.8016</u>	<u>0.5049</u>	<u>0.6735</u>	0.3922
	w/o Pretrain	0.8439	0.3949	0.8008	0.5057	0.6727	<u>0.3917</u>
	w/o Both	0.8438	0.3960	0.7993	0.5073	0.6706	0.3966
AutoInt	ClickPrompt	<b>0.8465</b>	<b>0.3912</b>	<b>0.8004</b>	<b>0.5076</b>	<b>0.6760</b>	<b>0.3924</b>
	w/o Prompt	0.8443	0.3992	<u>0.7999</u>	<u>0.5082</u>	<u>0.6722</u>	0.3985
	w/o Pretrain	<u>0.8450</u>	<u>0.3953</u>	0.7987	0.5092	0.6720	<u>0.3945</u>
	w/o Both	0.8448	0.3967	0.7982	0.5127	0.6699	0.3978
DNN	ClickPrompt	<b>0.8455</b>	<b>0.3909</b>	<b>0.7994</b>	<b>0.5080</b>	<b>0.6742</b>	<b>0.3935</b>
	w/o Prompt	0.8437	0.3959	<u>0.7988</u>	<u>0.5079</u>	0.6699	0.3979
	w/o Pretrain	<u>0.8445</u>	<u>0.3951</u>	0.7972	0.5123	<u>0.6718</u>	<u>0.3947</u>
	w/o Both	0.8441	0.3953	0.7973	0.5128	<u>0.6698</u>	<u>0.3996</u>

通过这些对比实验，我们旨在量化ClickPrompt的优势，即通过PA-MLM的软提示带来的对语义和协同信息的精细融合。

这些观察体现在表格中。当去除提示接口或PA-MLM预训练时，无论在DCNv2、AutoInt和DNN这类基础CTR模型上，还是在MovieLens-1M、BookCrossing和Amazon-Toys三个数据集上，性能都显著下降。这表明，明确的协作知识与语义信息的交互和精确的对齐是提取和整合输入信息的关键，这对提升CTR预测的准确性至关重要。



在长尾用户/项目领域，我们探究了ClickPrompt相对于基础CTR模型性能提升的深层次原因。以DCNv2为基线，我们使用RoBERTa-base作为PLM，针对MovieLens-1M数据集进行了实验，采用finetune-with-PLM策略。实验步骤如下：

- 1. 长尾效应<sup>+</sup>：结果显示，ClickPrompt显著优于DCNv2处理长尾用户/项目，这证明它能更有效地利用PLM的语义资源，特别是在资源稀缺或新领域初始化的条件下。
- 2. 信息融合：通过PA-MLM的软提示预训练，ClickPrompt能精巧地结合PLM的语义和自身的协同信息，显著提升预测准确度。
- 3. 适应性：即使面对非长尾用户/项目，ClickPrompt仍表现出一定优势，表明其具有良好的跨域泛化能力。
- 4. 相对提升：总体上，ClickPrompt在所有观测点上都显示出明显的性能提升，证实了其在提升CTR预测时的有效性和优越性。

**Table 6: The performance of DCNv2 and ClickPrompt (DCNv2 as backbone) for the long-tail user/item problems on MovieLens-1M dataset. The best results are given in bold Rel.Impr denotes the relative AUC improvement rate.**

Long-tail User ?	Long-tail Item ?	DCNv2		ClickPrompt		Rel.Impr
		AUC	Log Loss	AUC	Log Loss	
✓	✓	0.6000	0.6624	<b>0.6500</b>	<b>0.6038</b>	8.33%
✗	✓	0.6886	0.6930	<b>0.7003</b>	<b>0.6888</b>	1.70%
✓	✗	0.8149	0.3977	<b>0.8186</b>	<b>0.3916</b>	0.45%
✗	✗	0.8485	0.3978	<b>0.8520</b>	<b>0.3926</b>	0.41%

原文《ClickPrompt: CTR Models are Strong Prompt Generators for Adapting Language Models to CTR Prediction》

发布于 2024-04-18 10:28 · IP 属地北京

华为 LLM ctr



理性发言，友善互动

2 条评论

默认 最新



划水怪

这篇论文是不是没把代码开源？

07-09 · 北京

回复 喜欢



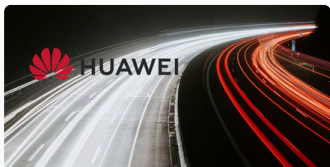
美感带来的好处

[github.com/CHIANGEL/Cli...](https://github.com/CHIANGEL/Cli...)

08-16 · 江苏

回复 喜欢

推荐阅读



极致安卓—手机Termux/Aid Learning组集群分布式并行...



如何运行plink软件--三和